COMPUTATION THEORY ON TWO TYPES

by

Johan Moldestad
Oslo 1976

Contents:

Abstract.

Recursion in higher types was first studied by Kleene. In his fundamental paper "Recursive functionals and quantifiers of finite type I, II" [12] he introduced nine schamata which generate the partial recursive functions of higher types. The domain of a partial recursive function is a subset of a cartesian product $Tp(i_1) \times \ldots \times Tp(i_n)$ where $i_1 \ldots i_n \in \omega$. $Tp(i)$ is defined by: $Tp(0) = \omega$, $Tp(n+1)$ is the set of functions from $Tp(n)$ to ω. A great deal of effort has been devoted to the study of these functions, and functions closely related to them, in particular the set $\{ \varphi :$ For some φ' which is partial recursive $\varphi = \lambda a\, \varphi'(a,F) \}$, where a is a variable ranging over some cartesian product of types $\leq n$, and F is a fixed object to type $n+2$.

The case $F = {}^2E$ has been of particular interest. The nine schemata for recursion in ${}^2E$ generate a hierarchy for the Δ_1^1 subsets of ω ([12]). The case F is of type 2 and normal (i.e. ${}^2E$ is recursive in F) also generates a nice hierarchy ([22]). Many of the deeper properties of recursion in a normal type-2 functional is due to R. Gandy [4], in particular the prewellordering property of computations and the existence of a selection operator.

There is a basic difference between the two cases $F = {}^2E$ and $F = {}^3E$. The last case was first studied by Moschovakis [14], and it was he that first observed that the relations which are recursively enumerable in ${}^3E$ are not closed under existantial quantifiers ranging over ${}^\omega\omega$. They are, however, closed under numerical quantifiers.

The case where F is of type $n+2$, $n > 0$, and F is normal

(i.e. ${}^{n+2}E$ is recursive in F), is in many ways similar to the case $F = {}^{3}E$. After some poineering work by Grilliot in his thesis and subsequent papers [5], [6], the case has recently been extensively studied in the theses of Harrington [7] and MacQueen [13]. In [13] it is proved that the relations which are recursively enumerable in F are not closed under existential quantifiers ranging over $Tp(n)$, but they are closed under existential quantifiers ranging over $Tp(n-1)$. So in this connection $Tp(n)$ plays the same role as the reals do in recursion in ${}^{3}E$. $Tp(n-1)$ plays the role of the numbers. It is natural to introduce the notions individuals and subindividuals. The individuals are the objects of type n, and the subindividuals are the objects of type $n-1$.

Recursion theory in a normal type 2 object has successfully been characterized in an "abstract" way, either in form of Spector theories or in form of Spector classes. The purpose of this paper is to study recursion in higher types as a recursion theory on a domain consisting of two types, viz. the subindividuals and the individuals. A similar approach has been adopted by Harrington and MacQueen in [10]. In the first part of the paper (§§ 1 - 5) we study recursion in a normal type 2 function over a domain of the form $A = S \cup S_{\omega}$, where S is the set of subindividuals and S_{ω} is the set of individuals. We shall see that most of the results of recursion in normal objects of type $n+2$, $n > 0$, hold in this setting, and that our results specialize to the classical ones when we take $S = Tp(0) \cup \ldots \cup Tp(n-1)$.

In the last part of the paper (§§ 6 - 9) we start out with a computation theory on a domain of two types. Computation theories were first studied by Moschovakis [16] and further developed by

Fenstad in [2], [3], and we refer to these papers for a survey of the general theory. The aim of this part is to prove the general plus 1 and plus 2 theorem in this setting of two types. For recursion in higher types these results are due to Sacks [21] and Harrington [7]. We also give a characterization of those computation theories which are equivalent to recursion in a normal type 2 functional over the domain.

We now present a more detailed outline of the paper.

§ 1 The computation domain

The set of subindividuals is denoted by S. We suppose that there is a tuple $\mathcal{S} = (N,s,M,K,L)$ such that $N \subseteq S$ is a copy of the natural numbers with successor function s. M is an injection $S \times S \to S$ (i.e. a pairing function), and K and L are the inverse functions of M, i.e. $K(M(r,s)) = r$, $L(M(r,s)) = s$. From these functions one can define an injection $\bigcup_{n<\omega} S^n \to S$ and a decoding function such that $(\langle x_1 \ldots x_k \rangle)_i = x_i$.

The set of individuals is the set ${}^S\omega$, the set of functions from S to ω. Let $A = S \cup {}^S\omega$. In a simple way the functions $M,K,L,\langle\ \rangle,(\cdot)$. can be extended from S to A.

Let $\mathfrak{A} = (A,S,\mathcal{S})$. This is our computation domain. A <u>functional</u> is a function F from A_S to ω, where A_S denotes the set of functions from A to S.

§ 2 Recursion on $\mathfrak{A}$

Let $\mathcal{L}$ be the list $R_1 \ldots R_k$, $\varphi_1 \ldots \varphi_l$, $F_1 \ldots F_m$ where $R_1 \ldots R_k$ are relations on A, $\varphi_1 \ldots \varphi_l$ are partial functions and $F_1 \ldots F_m$ are functionals. We define the functions which are partial recursive in $\mathcal{L}$ by 17 schemata. They are similar to the 9 schemata of Kleene in [12], and to those introduced by Moschovakis in [16]. We define a monotone inductive operator $\Gamma : \bigcup_{n<\omega} A^n \to \bigcup_{n<\omega} A^n$. If $(e,a,r) \in \Gamma(X)$ then $e \in N$, a is a list of length n of objects from A, $r \in S$. The function $\{e\}^{\mathcal{L}}$ is defined by: $\{e\}^{\mathcal{L}}(a) \simeq r$ iff $(e,a,r) \in \Gamma^{\infty}$. f is <u>partial recursive in $\mathcal{L}$</u> if $f = \{e\}^{\mathcal{L}}$ for some e. In this case e is an <u>index</u> for f. Some of the clauses in the definition of Γ are given below:

I $\quad (\langle 1,n+1\rangle, x,a,0) \in \Gamma(X)$ if $x \in N$,
$\quad (\langle 1,n+1\rangle, x,a,1) \in \Gamma(X)$ if $x \notin N$.

By this clause the characteristic function of N is recursive. Clauses II - VIII take care of the characteristic function of S, the functions $f(a) = m$ where $m \in N$, the functions s,M,K,L.

IX $\quad (\langle 9,n+2\rangle, x,y,a,x(y)) \in \Gamma(X)$ if $x \in {}^{S}\omega$, $y \in S$
$\quad (\langle 9,n+2\rangle, x,y,a,0) \in \Gamma(X)$ otherwise.

X $\quad$ If $\exists y[(e,a,y) \in X$ and $(f,y,a,x) \in X]$
$\quad$ then $(\langle 10,n,e,f\rangle, a,x) \in \Gamma(X)$.

By clause IX the function $f(x,y) = x(y)$ if $x \in {}^{S}\omega$, $y \in S$, $f(x,y) = 0$ otherwise, is recursive. By X the partial recursive functions are closed under substitution. The clauses XI and XII are for primitive recursion and permutations of the list of arguments. By clause XIII there is a partial recursive function which enumerates all partial recursive functions.

XIII $\quad$ If $(e,a,x) \in X$ then $(\langle 13,n+1\rangle, e,a,x) \in \Gamma(X)$.

The xlauses XIV - XVI are for the list $\mathcal{L}$.

XIV $(\langle 14, j_i+n, i\rangle,\ b,a,0) \in \Gamma(X)$ if $R_i(b)$

$(\langle 14, j_i+n, i\rangle,\ b,a,1) \in \Gamma(X)$ if $\neg R_i(b)$

$i = 1 \ldots k$, b has length j_i.

XVI If $\forall x\, \exists y (e,x,a,y) \in X$ then

$(\langle 16,n,e,i\rangle,\ a,\ F_i(f)) \in \Gamma(X)$ where

$f(x) = y$ iff $(e,x,a,y) \in X$, $i = 1 \ldots m$.

XVII If $\forall x \in S$ $\exists y(e,x,a,y) \in X$ and

$(e',z,a,u) \in X$ then $(\langle 17,n,e,e'\rangle,a,u) \in \Gamma(X)$, where

$z \in {}^{S}\omega$ is defined by: $z(x) = y$ iff $(e,x,a,y) \in X$.

The functionals are introduced in XVI. Clause XVII says that if $z \in {}^{S}\omega$ is recursive in $\mathcal{L}$, and z occurs in an argument list, then z can be taken away from the list. Instead its index is introduced.

The functions which are primitive recursive can be obtained by omitting clause XIII and the clauses for the list $\mathcal{L}$.

A <u>convergent computation</u> in $\mathcal{L}$ is a tuple $\langle e,a\rangle$ such that $\{e\}^{\mathcal{L}}(a)\downarrow$ (i.e. $\{e\}^{\mathcal{L}}(a) \simeq r$ for some r). To each convergent computation $\langle e,a\rangle$ we associate an ordinal $|\langle e,a\rangle|^{\mathcal{L}}$ = the least μ such that $\exists r(e,a,r) \in \Gamma^{\mu+1}$. Sometimes this ordinal is denoted by $|\{e\}^{\mathcal{L}}(a)|^{\mathcal{L}}$. Let $\kappa^{\mathcal{L}}$ be the closure ordinal of Γ.

Suppose $\{e\}^{\mathcal{L}}(a)\downarrow$. There is a natural way to define the subcomputations of $\langle e,a\rangle$. The set of subcomputations of $\langle e,a\rangle$ is recursive in $\mathcal{L}$ and $\langle e,a\rangle$, uniformly in $\langle e,a\rangle$ when $\{e\}^{\mathcal{L}}(a)\downarrow$. From the subcomputations of $\langle e,a\rangle$ one can construct the <u>computation tree</u> of $\langle e,a\rangle$.

The definition of a subcomputation of $\langle e,a\rangle$ can be extended to arbitrary $\langle e,a\rangle$. The set of subcomputations of $\langle e,a\rangle$ is recursively enumerable in $\mathcal{L}$, $\langle e,a\rangle$, uniformly in $\langle e,a\rangle$.

Theorem 1: For all e,a : $\{e\}^{\mathcal{L}}(a)\downarrow$ iff the computation tree of $\langle e,a\rangle$ is wellfounded.

Computation trees have been studied in [12] and [13].

§ 3 Connection with Kleene recursion in higher types

In this chapter $S = Tp(0) \cup \ldots \cup Tp(n-1)$, where $n > 0$. Then S_ω can be identified with $Tp(1) \times \ldots \times Tp(n)$. Suppose F is an object to type $n+2$. We prove that there is a list $\mathcal{L}$ of relations, functions and functionals such that recursion in F on $Tp(0),\ldots,Tp(n)$ is essentially the same as recursion in $\mathcal{L}$ on $\mathfrak{A}$. The opposite is also true. If $\mathcal{L}$ is a list which contains relations, functions and functionals expressing the type structure of S, then there are objects of $Tp(n+1)$ and $Tp(n+2)$ such that recursion in $\mathcal{L}$ on $\mathfrak{A}$ is essentially the same as recursion in these objects on the types.

§ 4 Recursion in normal lists on $\mathfrak{A}$

Let E be the functional defined by:

$$E(f) = \begin{cases} 0 & \text{if } \exists x\, f(x) = 0 \\ 1 & \text{if } \forall x\, f(x) \neq 0 \end{cases}$$

where $f \in A_S$. A list $\mathcal{L}$ is normal if the equality relation on S is recursive in $\mathcal{L}$, and E is weakly recursive in $\mathcal{L}$. A functional F is recursive in $\mathcal{L}$ if there is an index e such that $F(f) = \{e\}^{\mathcal{L},f}(0)$ for all f. F is weakly recursive in $\mathcal{L}$ if there is a primitive recursive function $s(e)$ such that for all e,a : $\{s(e)\}^{\mathcal{L}}(a) \simeq F(\lambda x\{e\}^{\mathcal{L}}(x,a))$. Moreover if $\{s(e)\}^{\mathcal{L}}(a)\downarrow$ then

$|\{s(e)\}^{\mathcal{L}}(a)| > |\{e\}^{\mathcal{L}}(x,a)|$ for all x.

Let $C^{\mathcal{L}}$ be the set of convergent computations.

<u>Theorem 2</u>: Let $\mathcal{L}$ be normal. There is a function p which is partial recursive in $\mathcal{L}$, such that $p(x,y)\downarrow$ if $x \in C^{\mathcal{L}}$ or $y \in C^{\mathcal{L}}$, in which case $p(x,y) \simeq 0$ if $|x|^{\mathcal{L}} \leq |y|^{\mathcal{L}}$, $p(x,y) \simeq 1$ if $|x|^{\mathcal{L}} > |y|^{\mathcal{L}}$.

<u>Theorem 3</u>: Let $\mathcal{L}$ be normal. There is a function φ which is partial recursive in $\mathcal{L}$ such that for all e,a: If $\exists n \in N\{e\}^{\mathcal{L}}(n,a)\downarrow$ then $\varphi(e,a)\downarrow$ and $\{e\}^{\mathcal{L}}(\varphi(e,a),a)\downarrow$. Moreover if $\varphi(e,a) \simeq n$ then $\{e\}^{\mathcal{L}}(n,a)\downarrow$.

Theorem 2 corresponds to theorem 6.1 in [13]. Theorem 3 is a corollary of theorem 2.

Suppose Y is a set of elements in ${}^{S}\omega$, indexed by S, i.e. $Y = \{\alpha_r : r \in S\}$. Then all elements in Y can be coded by one element in ${}^{S}\omega$, namely α defined by: $\alpha(r) = \alpha_{(r)_1}((r)_2)$. This is utilized in theorem 4 (corollary 5.2 in [13]).

<u>Theorem 4</u>: There is a relation R which is recursively enumerable in $\mathcal{L}$ such that for all e,a: $\{e\}^{\mathcal{L}}(a)\uparrow \iff \exists \alpha R(\alpha, \langle e,a\rangle)$.

<u>Corollary</u>: The relations which are recursively enumerable in $\mathcal{L}$ are not closed under existential quantifiers over ${}^{S}\omega$.

<u>Theorem 7</u> (Grilliot's selection theorem): Let $\mathcal{L}$ be normal. Let $B \subseteq S$ be recursively enumerable in $\mathcal{L}, a$; $B \neq \emptyset$. Then there is a subset B' of B such that B' is recursive in $\mathcal{L}, a$, and $B' \neq \emptyset$.

(Theorem 7.2 in [13].)

Corollary: The relations which are recursively enumerable in $\mathcal{L}$ are closed under existential quantifiers ranging over S.

§ 5 Kleene-recursion in normal objects of type $n+2$, $n>0$

We use the equivalence result from § 3 to see that results about recursion in higher types can be deduced from corresponding results about recursion on $\mathfrak{A}$.

§ 6 Computation theories on $\mathfrak{A}$

A computation theory on $\mathfrak{A}$ is a pair $(\Theta, |\,|_\Theta)$. Θ is a set of tuples (e,a,r) where $e \in N$, a is a list of objects from A, $r \in S$. $|\,|_\Theta$ is a function from Θ onto some ordinal $\varkappa$. φ is Θ-computable if $\varphi(a) \simeq r$ iff $(e,a,r) \in \Theta$ for some e. φ is denoted by $\{e\}_\Theta$. For basic definitions and general results about computation theories we refer to Moschovakis [16] or to the survey papers of Fenstad [2], [3]. We emphaxise that our computation theories always are singlevalued.

Let X be a subset of A such that X is Θ-computable. X is strongly Θ-finite if the partial functional $\mathcal{F}_X$ defined by

$$\mathcal{F}_X(\varphi) = \begin{cases} 0 & \text{if } \exists x \in X \ \ \varphi(x) \simeq 0 \\ 1 & \text{if } \forall x \in X \ \ \exists r \neq 0 \ \ \varphi(x) \simeq r \end{cases}$$

is weakly Θ-computable. X is weakly Θ-finite if the functional F_X defined by

$$F_X(f) = \begin{cases} 0 & \text{if } \exists x \in X \ \ f(x) = 0 \\ 1 & \text{if } \forall x \in X \ \ \exists r \neq 0 \ \ f(x) = r \end{cases}$$

is weakly Θ-computable. ($\mathcal{F}_X$ is defined on partial functions, F_X on total functions.)

Let $C_\Theta = \{\langle e,a\rangle : \{e\}_\Theta(a)\downarrow\}$. Θ is <u>p-normal</u> if there is a Θ-computable function $p(x,y)$ such that $p(x,y)\downarrow$ if $x \in C_\Theta$ or $y \in C_\Theta$, in which case $p(x,y) = 0$ if $|x|_\Theta \leq |y|_\Theta$, $p(x,y) = 1$ if $|x|_\Theta > |y|_\Theta$.

<u>Lemma 14 and 15</u>: If Θ is p-normal, then Θ admits selection operators for numbers. If E is weakly Θ-computable then A is weakly finite. If $(\Theta, ||_\Theta)$ is recursion in a normal list, then S is strongly finite.

§ 7 Abstract Kleene theories

In this chapter we introduce the notion of an abstract recursion theory on $Tp(0),\ldots,Tp(n)$, and prove that this is essentially the same as an abstract recursion theory on $\mathfrak{A}$ when $S = Tp(0)\cup\ldots\cup Tp(n-1)$.

§ 8 Normal computation theories on $\mathfrak{A}$

Θ is <u>normal</u> if the equality relation on S is Θ-computable, A is weakly finite, S is strongly finite, and Θ is p-normal.

Suppose Θ is normal. There are some interesting ordinals associated to Θ :

$$\kappa = \sup\{|x|_\Theta : x \in C_\Theta\}$$

$$\kappa^0 = \sup\{|x|_\Theta : x \in C_\Theta \cap N\}$$

$$\kappa^a = \sup\{|\langle e,a\rangle|_\Theta : \langle e,a\rangle \in C_\Theta\}$$

$$\kappa^S = \sup\{|x|_\Theta : x \in C_\Theta \cap S\}$$

$$\kappa^{S,a} = \sup\{|\langle e,a,b\rangle|_\Theta : \langle e,a,b\rangle \in C_\Theta, e,b \in S\}.$$

The order relation between these ordinals is:

$$\kappa^0 \leq \kappa^a \leq \kappa^{S,a} < \kappa.$$

Definition: Let $\sigma \leq \kappa_{\Theta}$. Then σ is <u>a-reflecting</u> if for all $e \in N$: $\exists x |\{e\}_{\Theta}(x,a)| < \sigma \Rightarrow \exists x |\{e\}_{\Theta}(x,a)| < \kappa^a$.

Lemma 24: Suppose $x \in B \iff \{e\}_{\Theta}(x,a)\downarrow$. Then i), ii) and iii) are equivalent. i) There is a subset B' of B which is nonempty and Θ-computable in a, ii) $\exists x |\{e\}_{\Theta}(x,a)| < \kappa^a$. iii) $\exists x |\{e\}_{\Theta}(x,a)| < \kappa_r^a$.

Lemma 25: Let $B \subseteq A$ be Θ-computable. Then i), ii) and iii) are equivalent. i) B is strongly Θ-finite. ii) For all e,a if $\exists x \in B \ \{e\}_{\Theta}(x,a)\downarrow$ then $\exists x \in B \ |\{e\}_{\Theta}(x,a)| < \kappa^a$. iii) If C is a nonempty subset of B which is Θ-semicomputable in a then there is a nonempty subset C' of C which is Θ-computable in a.

The notion of reflection was first introduced by Harrington in [7]. The next theorem and the corollary are proved there.

Let a be fixed. Let $P = \{\langle e,b\rangle : \{e\}_{\Theta}(a,b)\downarrow$, b is a list of objects from $S\}$. $P \subset S$, hence $P \in {}^{S}\omega$. P is a complete Θ-semicomputable in a subset of S. $\kappa^{S,a} < \kappa^{a,P}$.

Theorem 8: $\kappa^{S,P,a}$ is a-reflecting.

Corollary: Suppose B is a set of subsets of S, such that B is Θ-semicomputable in a, and B contains an element which is Θ-semicomputable in a. Then B contains an element which is Θ-computable in a.

Definitions:

$$\begin{aligned}
sc(\Theta) &= \{X \subseteq A : X \text{ is } \Theta\text{-computable}\},\\
sc(\Theta,a) &= \{X \subseteq A : X \text{ is } \Theta\text{-computable in } a\},\\
en(\Theta) &= \{X \subseteq A : X \text{ is } \Theta\text{-semicomputable}\},\\
S\text{-}en(\Theta) &= \{X \subseteq S : X \text{ is } \Theta\text{-semicomputable}\}.
\end{aligned}$$

<u>Theorem 9</u>: Let Θ be normal. Then there is a normal list $\mathcal{L}$ such that S-en(Θ) = S-en($\mathcal{L}$), and for all $r \in S : sc(\Theta,r) = sc(\mathcal{L},r)$.

This is an abstract version of the plus 2 theorem in Harrington [7] and also of the plus 1 theorem of Sacks [21]. The original plus 2 theorem of Harrington was a reduction result: Starting out with a normal functional G of type > n+2 he constructed a functional F of type n + 2 such that ${}_n\mathrm{en}(G) = {}_n\mathrm{en}(F)$. This fact uses the fact that Tp(n) is strongly finite in G. Theorem 9 is an improvement in the sense that we start out with a normal computation theory Θ. Hence in the concrete setting of higher types we only assume that Tp(n) is weakly Θ-finite whereas Tp(n-1) is strongly Θ-finite. Thus theorem 9 gives a kind of characterization result. The proof is quite similar to Harrington's proof in [7]. However, some modifications are necessary, and I am grateful to L. Harrington for helpful suggestions in this connection.

In the last part of chapter 8 there is a characterization of those computation theories which are equivalent to recursion in a normal list. We consider computation theories $(\Theta, ||_\Theta)$ on $\mathfrak{A}$ such that the equality relation on S is Θ-computable, E is weakly recursive in Θ, and Θ is p-normal. This is weaker than normality: We do not suppose that S is strongly finite.

<u>Definition</u>: En(Θ) = $\{\varphi : \varphi$ is Θ-computable$\}$.

<u>Definition</u>: Θ is <u>Mahlo</u> if for all normal lists $\mathcal{L}$:
$\mathcal{L}$ is Θ-computable $\Rightarrow \exists x(\kappa^x_{\mathcal{L}} < \kappa^x_\Theta)$.

<u>Theorem 10</u>: Let Θ satisfy the properties mentioned above. Then Θ is not Mahlo iff there is a normal Θ-computable list $\mathcal{L}$ such that En($\mathcal{L}$) = En(Θ).

References: Thm. 3.2 in [11]. When Θ is a computation theory on ω this is also proved in [8] and [9]. A different characterization theorem has been developed by D. Normann [19], using his imbedding theory in higher types.

§ 9 More about Maholoness

In ordinal recursion the notion of Maholness is defined in the following way: An ordinal τ is Mahlo if τ is recursively regular, and all normal functions π which are τ-recursive in constants less that τ have a recursively regular fixed point less that τ. (Definition 4.2 (b) in [1].) The purpose of this chapter is to prove that the definition of Mahloness given in § 8 is a natural generalization of the definition above.

To see this let us regard normal computation theoreis $(\Theta, ||_\Theta)$ with domain ω, i.e. ω is strongly Θ-finite and $||_\Theta$ is a Θ-norm. Then $\kappa_\Theta^x = \kappa_\Theta$ for all $x \in \omega$. The analogue of the notion of Mahloness as defined in § 8 is the following: Θ is Mahlo$_1$ if $\kappa^{\mathcal{L}} < \kappa_\Theta$ for all normal lists $\mathcal{L}$ which are Θ-computable. The notion of Mahloness can also be defined in a way which is more similar to the definition above. In theorem 11 we prove that the two notions of Mahoness are equivalent.

It is the second notion of Maholness which has been generalized by Dag Normann in his study [19]. From the characterization theorem in § 8 it follows that this notion is in some sense equivalent to our definition in § 8. A direct equivalence proof similar to the proof of theorem 11 has not yet been provided.

Acknowledgements

Professor Jens Erik Fenstad inspired me to write this paper. He has given many valuable comments during the preparation concerning selection of topics, disposition, improvements etc. I want to thank him for this, and for the interest he has shown.

§ 1 THE COMPUTATION DOMAIN

Let S be an infinite set. A <u>coding scheme</u> for S is a quintuple $\mathcal{S} = (N,+1,M,K,L)$ where $N \subseteq S$ is a copy of the natural numbers with successor function $+1$. M is an injection $S \times S \to S$, i.e. for all $r,s,r',s' \in S$: $M(r,s) = M(r',s') \Rightarrow r = r'$ and $s = s'$. K and L are functions $S \to S$ such that for all $r,s \in S$: $K(M(r,s)) = r$, $L(M(r,s)) = s$. In addition N is closed under K, L and M.

There are some functions and a predicate associated to $\mathcal{S}$.

For each natural number n there is an injection $\langle\ \rangle_n : S^n \to S$ defined by:

$$\langle\ \rangle_0 = 0$$

$$\langle r_1,\dots,r_{n+1}\rangle_{n+1} = M(\langle r_1,\dots,r_n\rangle_n, r_{n+1}) .$$

$\langle\ \rangle$ is an injection $\bigcup\limits_{n<\omega} S^n \to S$ defined by:

$$\langle r_1,\dots,r_n\rangle = M(n,\langle r_1,\dots,r_n\rangle_n) .$$

The predicate Seq is the image of $\langle\ \rangle$, i.e.

$$\mathrm{Seq}(r) \iff \exists n\ \exists r_1\dots r_n \quad r = \langle r_1\dots r_n\rangle .$$

The elements of Seq are called sequences, and there is a function $\mathrm{lh} : S \to N$ which gives the length of sequences:

$$\mathrm{lh}(r) = \begin{cases} 0 & \text{if} \quad \mathrm{Seq}(r) \\ K(r) & " \quad \mathrm{Seq}(r) \end{cases}$$

$\lambda r i\, (r)_i$ is a function $S \times N \to S$ such that $(\langle r_1 \dots r_i \dots r_n\rangle)_i = r_i$.

$$(r)_i = \begin{cases} 0 & \text{if} \quad (\mathrm{Seq}(r) \text{ and } i \in N \text{ and } i \le \mathrm{lh}(r)) \\ r_i & \text{if} \quad r = \langle r_1 \dots r_i \dots r_n\rangle . \end{cases}$$

Let $A = S \cup S_\omega$, where S_ω is the set of functions from S to ω. (ω is the set of natural numbers. ω and N will not be distinguished.)

Notations:

Elements in N, ω : e,f,g,h,i,j,k,l,m,n.

Elements in S : r,s.

Elements in S_ω : $\alpha,\beta,\gamma,\delta$.

Elements in A : x,y,z,u,v,w.

Finite lists of elements from A : a,b,c,d.

Total functions $A^n \to S$: f,g,h.

Partial functions $A^n \to S$: φ,ψ.

Relations on A^n : R.

Total functionals : F,G

Partial functionals : $\mathcal{F}$ (in §8 also G_τ).

Ordinals : $\epsilon,\eta,\varkappa,\lambda,\mu,\nu,\xi,\pi,\rho,\sigma,\tau$.

In §9 the letters π,ρ,σ are reserved for ordinal functions.

Computation theories: $(\Theta,||_\Theta)$, $(\Psi,||_\Psi)$.

<u>Definition</u>: The triple $(A,S,\mathcal{S})$ is called a <u>computation domain</u>. A is the <u>universe</u> of the computation domain. S is the set of <u>subindividuals</u>.

Let $*$ be the injection $S \to S_\omega$ defined by:

$$r^*(s) = \begin{cases} 0 & \text{if } s = r \\ 1 & \text{"} \quad s \neq r \end{cases}$$

Let $^-$ be a function $A \to B$ defined by:

$$x^- = \begin{cases} r & \text{if } x = r^* \\ 0 & \text{if } x \text{ is not in the image of } * \end{cases}$$

Let $\mathcal{S}$ be a coding scheme for S. It is possible to extend the functions M, K, L to A and hence derive a coding scheme for A, for instance in the following way:

$$M(r,\alpha) = \lambda s\, M(M(r^*(s),\alpha(s)),M(0,1))$$

$$M(\alpha,r) = \lambda s\, M(M(\alpha(s),r^*(s)),M(1,0))$$

$$M(\alpha,\beta) = \lambda s\, M(M(\alpha(s),\beta(s)),M(1,1))$$

$M(r,\alpha)$, $M(\alpha,r)$, $M(\alpha,\beta)$ are elements of S_ω because N is closed under M. Let

$$\begin{aligned} K(\alpha) &= (\lambda s\, K \circ K(\alpha(s)))^- && \text{if } L(\alpha(0)) = M(0,1) \\ &= \lambda s\, K \circ K(\alpha(s)) && \text{otherwise} \end{aligned}$$

$$\begin{aligned} L(\alpha) &= (\lambda s\, L \circ K(\alpha(s)))^- && \text{if } L(\alpha(0)) = M(1,0) \\ &= \lambda s\, L \circ K(\alpha(s)) && \text{otherwise} \end{aligned}$$

The extended functions M, K, L have the properties:

$\forall x\, y\, x'\, y' : M(x,y) = M(x',y') \Longrightarrow x = x'$ and $y = y'$,

$\forall x\, y : M(x,y) \in S \Longleftrightarrow x \in S$ and $y \in S$,

$\forall x\, y : K(M(x,y)) = x$, $L(M(x,y)) = y$.

Obviously the functions $\langle\ \rangle_n (n \in \omega)$, $\langle\ \rangle$, lh, $\lambda x\, i (x)_i$ and the predicate Seq can be extended to A since they are defined from M, K, L.

§ 2 RECURSION ON $\mathfrak{A}$

Let the triple $(A, S, \mathscr{S})$ be denoted by $\mathfrak{A}$. There is a natural class of functions associated to $\mathfrak{A}$: the class of primitive recursive functions on $\mathfrak{A}$, denoted by PRF.

It is the smallest class of functions containing:

$$f(x,a) = \begin{cases} 0 & \text{if} \quad x \in N \\ 1 & \text{"} \quad x \notin N \end{cases}$$

$$f(x,a) = \begin{cases} 0 & \text{if} \quad x \in S \\ 1 & \text{"} \quad x \notin S \end{cases}$$

$$f(x,a) = \begin{cases} x & \text{if} \quad x \in S \\ 0 & \text{"} \quad x \notin S \end{cases}$$

$$f(x,a) = \begin{cases} x+1 & \text{if} \quad x \in N \\ 0 & \text{"} \quad x \notin N \end{cases}$$

$$f(a) = m \qquad (m \in N)$$

$$f(x,y,a) = \begin{cases} M(x,y) & \text{if} \quad x,y \in S \\ 0 & \text{otherwise} \end{cases}$$

$$f(x,a) = \begin{cases} K(x) & \text{if} \quad x \in S \\ 0 & \text{"} \quad x \notin S \end{cases}$$

$$f(x,a) = \begin{cases} L(x) & \text{if} \quad x \in S \\ 0 & \text{"} \quad x \notin S \end{cases}$$

$$f(x,y,a) = \begin{cases} x(y) & \text{if} \quad x \in {}^{S}\omega \quad \text{and} \quad y \in S \\ 0 & \text{otherwise} \end{cases}$$

and with the following closure properties:

if $f,g \in PRF$ then $h \in PRF$ where h is defined by:

i) $h(a) = f(g(a),a)$,

ii) $h(0,a) = f(a)$

$h(n+1,a) = g(h(n,a),a,n)$

$h(x,a) = 0$ if $x \notin N$,

iii) $h(a) = f(a')$ where a' is a permutation of the list a .

Let $R_1 \dots R_k$ be predicates, $f_1 \dots f_l$ functions with values in S and $F_1 \dots F_m$ functionals. (A <u>functional</u> is a total function: $T \to \omega$, where T is the set of total functions from A to S .) The class functions which are <u>primitive recursive in $R_1 \dots R_k$, $f_1 \dots f_l$, $F_1 \dots F_m$</u> is obtained by adding the following clauses:

$$f(a,b) = \begin{cases} 0 & \text{if } R_i(a) \\ 1 & \text{"} \ \neg R_i(a) \end{cases} \qquad i = 1 \dots k ,$$

$f(a,b) = f_i(a)$, $i = 1 \dots l$,

$f(a) \quad = F_i(\lambda x \; g(x,a))$, $i = 1 \dots m$,

$h(a) \quad = f(\lambda r \; g(r,a),a)$.

where g has values in N .

With these clauses one can substitute a function for an element in ${}^S\omega$, i.e. if f,g are primitive recursive in $R_1 \dots R_k$, $f_1 \dots f_l$, $F_1 \dots F_m$ then so are h where $h(a) = f(\lambda r \; g(r,a),a)$.

<u>Lemma 1</u>: The graphs of the functions $*$, $^-$, of the extended functions $\langle \ \rangle_n$, lh , $\lambda x i(x)_i$, and the extended predicate Seq are primitive recursive in the equality relation on S and the functional E_S defined below.

$$E_S(f) = \begin{cases} 0 & \text{if} \quad \exists s \in S \quad f(s) = 0 \\ 1 & \text{"} \quad \forall s \in S \quad f(s) \neq 0 \end{cases}$$

where $f : A \to S$ is total.

In [12] Kleene has defined the class of partial recursive functions on the pure types. In [15], [16] Moschovakis defined the class of prime computable functions. Here the class of partial recursive functions on $\mathfrak{A}$ is defined in an analoguous way. A set of computations is defined inductively by the operator Γ which is given below. In the definition of Γ there is one clause for each of the functions and closure properties which defined the primitive recursive functions. In addition there is one clause for diagonalization (clause XIII).

Let $X \subseteq \bigcup_{n<\omega} A^n$. $\Gamma(X)$ is the subset of $\bigcup_{n<\omega} A^n$ defined by:

For all $n \in N$, all lists a of length n

I $\quad ((\langle 1,n+1\rangle,x,a,0) \in \Gamma(X)$ if $x \in N$

$\quad ((\langle 1,n+1\rangle,x,a,1) \in \Gamma(X)$ " $x \notin N$

II $\quad ((\langle 2,n+1\rangle,x,a,0) \in \Gamma(X)$ if $x \in S$

$\quad ((\langle 2,n+1\rangle,x,a,1) \in \Gamma(X)$ if $x \notin S$

III $\quad ((\langle 3,n+1\rangle,x,a,x) \in \Gamma(X)$ if $x \in S$

$\quad ((\langle 3,n+1\rangle,x,a,0) \in \Gamma(X)$ if $x \notin S$

IV $\quad ((\langle 4,n+1\rangle,x,a,x+1) \in \Gamma(X)$ if $x \in N$

$\quad ((\langle 4,n+1\rangle,x,a,0) \in \Gamma(X)$ if $x \notin N$

V $\quad ((\langle 5,n,m\rangle,a,m) \in \Gamma(X)$

VI $\quad ((\langle 6,n+2\rangle,x,y,a,M(x,y)) \in \Gamma(X)$ if $x,y \in S$

$\quad ((\langle 6,n+2\rangle,x,y,a,0) \in \Gamma(X)$ otherwise

VII $\quad ((\langle 7,n+1\rangle,x,a,K(x)) \in \Gamma(X)$ if $x \in S$

$\quad ((\langle 7,n+1\rangle,x,a,0) \in \Gamma(X)$ if $x \notin S$

VIII $((\langle 8,n+1\rangle,x,a,L(x)) \in \Gamma(X)$ if $x \in S$

$((\langle 8,n+1\rangle,x,a,0) \in \Gamma(X)$ if $x \notin S$

IX $((\langle 9,n+2\rangle,x,y,a,x(y)) \in \Gamma(X)$ if $x \in {}^{S}\omega$ and $y \in S$,

$((\langle 9,n+2\rangle,x,y,a,0) \in \in \Gamma(X)$ if $x \notin {}^{S}\omega$ or $y \notin {}^{S}\omega$

X If $\exists y[(e,a,y) \in X$ and $(e',y,a,x) \in X]$

then $((\langle 10,n,e,e'\rangle,a,x) \in \Gamma(X)$

XI If $(e,a,x) \in X$ then $((\langle 11,n+1,e,e'\rangle,0,a,x) \in \Gamma(X)$.

If $\exists y[((\langle 11,n+1,e,e'\rangle,m,a,y) \in X$

and $(e',y,m,a,x) \in X]$ then

$((\langle 11,n+1,e,e'\rangle,m+1,a,x) \in \Gamma(X)$

XII If $(e,a',x) \in X$ then $((\langle 12,n,e,i\rangle,a,x) \in \Gamma(X)$, where a' is obtained from a by moving the $i+1$-st object in a to the front of the list.

XIII If $(e,a,x) \in X$ then $((\langle 13,n+1\rangle,e,a,x) \in \Gamma(X)$.

Let $\mathcal{L}$ be the list $R_1 \dots R_k$, $\varphi_1 \dots \varphi_l$, $F_1 \dots F_m$ where $R_1 \dots R_k$ are predicates, $\varphi_1,,,\varphi_l$ are partial functions with values in S and $F_1 \dots F_m$ are functionals. The functions which are partial recursive in $\mathcal{L}$ are obtained by adding the following clauses to Γ:

XIV $((\langle 14,j_i+n,i\rangle,b,a,0) \in \Gamma(X)$ if $R_i(b)$

$((\langle 14,j_i+n,i\rangle,b,a,1) \in \Gamma(X)$ if $\neg R_i(b)$,

$i = 1 \dots k$, b has length j_i,

XV $((\langle 15,j_i+n,i\rangle,b,a,\varphi_i(b)) \in \Gamma(X)$ if $b \in \operatorname{dom}\varphi_i$, $i = 1 \dots l$,

XVI If $\forall x \exists y(e,x,a,y) \in X$ then

$((\langle 16,n,e,i\rangle,a,F_i(f)) \in \Gamma(X)$ where

$f(x) = y \iff (e,x,a,y) \in X$, $i = 1,\dots,m$.

XVII If $x \in S$ $y \in N(e,x,a,y) \in X$ and

$(e',z,a,u) \in X$ then

$((\langle 17,n,e,e'\rangle,a,u) \in \Gamma(X)$, where

$z \in {}^{S}\omega$ is defined by: $z(x) = y$ iff $(e,x,a,y) \in X$

Let $\Gamma^{0} = \emptyset$, $\Gamma^{\nu+1} = \Gamma^{\nu} \cup \Gamma(\Gamma^{\nu})$, $\Gamma^{\lambda} = \bigcup_{\nu<\lambda} \Gamma^{\nu}$ if $\lim \lambda$. $\Gamma^{\infty} = \bigcup_{\nu \in On} \Gamma^{\nu}$ (On = the ordinals.) Γ^{∞} is the set inductively defined by Γ.

For $e \in N$ let $\{e\}^{\mathcal{L}}$ be the partial function defined by: $\{e\}^{\mathcal{L}}(a) \simeq x$ iff $(e,a,x) \in \Gamma^{\infty}$. Then $\{e\}^{\mathcal{L}}$ is singlevalued. Let $\mathcal{P}(\mathcal{L}) = \{\{e\}^{\mathcal{L}} : e \in N\}$. $\mathcal{P}(\mathcal{L})$ is the set of functions which are <u>partial recursive in $\mathcal{L}$</u>. If clause XIII is removed from Γ one obtains the class of functions which are primitive recursive in $\mathcal{L}$.

Let $|\ |^{\mathcal{L}} : \Gamma^{\infty} \to On$ be the function defined by

$|e,a,x|^{\mathcal{L}}$ = the least μ such that $(e,a,x) \in \Gamma^{\mu+1}$

If $(e,a,x) \notin \Gamma^{\infty}$ let $|e,a,x| = \varkappa^{\mathcal{L}}$, where $\varkappa^{\mathcal{L}} = \sup\{|e,a,x|^{\mathcal{L}} : (e,a,x) \in \Gamma^{\infty}\}$.
$\varkappa^{\mathcal{L}}$ is a limit ordinal, and $|e,a,x|^{\mathcal{L}} < \varkappa$ for all $(e,a,x) \in \Gamma^{\infty}$.

<u>Computations and subcomputations</u>:

A <u>computation</u> is a tuple (e,a,x). It is <u>convergent</u> if $(e,a,x) \in \Gamma^{\infty}$. Otherwise it is <u>divergent</u>. If $(e,a,x) \in \Gamma^{\infty}$ then x is unique, i.e. $(e.a.x) \in \Gamma^{\infty}$ and $(e,a,x') \in \Gamma^{\infty} \Rightarrow x = x'$. Hence there is no ambiguity in denoting the computation by $\langle e,a\rangle$. Sometimes it will be denoted by $\{e\}^{\mathcal{L}}(a)$. (Hence $\{e\}^{\mathcal{L}}(a)$ has a double meaning: it denotes a computation, and also the object x such that $\{e\}^{\mathcal{L}}(a) \simeq x$.) Let $|\{e\}^{\mathcal{L}}(a)|^{\mathcal{L}} = |\langle e,a\rangle|^{\mathcal{L}} = |e,a,x|^{\mathcal{L}}$ where $\{e\}^{\mathcal{L}}(a) \simeq x$. If there is no x such that $\{e\}^{\mathcal{L}}(a) \simeq x$ let

$|\{e\}^{\mathcal{L}}(a)|^{\mathcal{L}} = |\langle e,a\rangle|^{\mathcal{L}} = \kappa^{\mathcal{L}}$.

Let "$\{e\}^{\mathcal{L}}(a)\downarrow$" be an abbreviation for the statement "there is an x such that $\{e\}^{\mathcal{L}}(a) \simeq x$". "$\{e\}^{\mathcal{L}}(a)\uparrow$" is an abbreviation for the negation of this statement.

Suppose $\{e_0\}^{\mathcal{L}}(a)\downarrow$. By looking at the definition of Γ we see that there is an obvious way to define the subcomputations of $\langle e_0,a\rangle$. First we define the <u>immediate subcomputations</u> (i.s.) of $\langle e_0,a\rangle$ by:

i) If $\{e_0\}^{\mathcal{L}}(a)\downarrow$ by one of the clauses I - IX, XIV, XV then there is no i.s. of $\langle e_0,a\rangle$.

ii) If $\{e_0\}^{\mathcal{L}}(a)\downarrow$ by clause X (substitution) then there are two i.s., namely $\langle e,a\rangle$ and $\langle e',\{e\}^{\mathcal{L}}(a),a\rangle$. If $\{e_0\}^{\mathcal{L}}(n,a)\downarrow$ by clause XI (primitive recursion) then $\langle e_0,0,a\rangle$ has one i.s., namely $\langle e,a\rangle$. $\langle e_0,m+1,a\rangle$ has two i.s., namely $\langle e_0,m,a\rangle$ and $\langle e',\{e_0\}^{\mathcal{L}}(m,a),m,a\rangle$.

iii) If $\{e_0\}^{\mathcal{L}}(a)\downarrow$ by the clauses XII or XIII then there is one i.s., namely $\langle e,a'\rangle$, $\langle e,a\rangle$ respectively.

iv) If $\{e_0\}^{\mathcal{L}}(a)\downarrow$ by Clause XVI then there is one i.s. for each $x \in A$, namely $\langle e,x,a\rangle$. If $\{e_0\}^{\mathcal{L}}(a)\downarrow$ by clause XVII then there is one i.s. for each $x \in S$, namely $\langle e,x,a\rangle$.

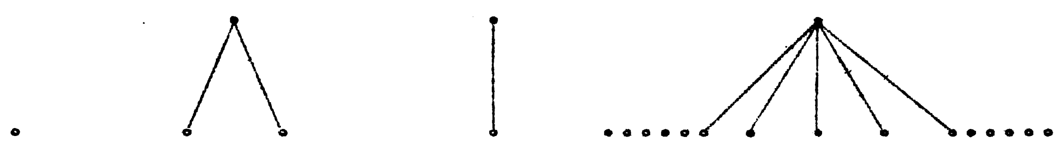
$\langle e',a'\rangle$ is a <u>subcomputation</u> of $\langle e,a\rangle$ if there is a finite sequence $x_0,x_1\ldots x_n$ such that $x_0 = \langle e,a\rangle$, $x_n = \langle e',a'\rangle$ and for $i = 0,1,\ldots,n-1$: x_{i+1} is an i.s. of x_i.

The relation "x is an i.s. of $\langle e_0,a\rangle$" can be defined for arbitrary $\langle e_0,a\rangle$ (the above definition applies only when $\{e_0\}^{\mathcal{L}}(a)\downarrow$.) i), iii) and iv) in the definition is changed as follows: "$\{e_0\}^{\mathcal{L}}(a)\downarrow$" is replaced by: "$e_0$ is an index corresponding to one of the clauses ..... and the length of the list a is the same as the number indicated by e_0 (i.e. the length of $a = (e_0)_2$)." ii) is replaced by ii'):

ii') If $e_0 = \langle 10,n,e,e'\rangle$ and the length of a is n then $\langle e,a\rangle$ is an i.s. of $\langle e_0,a\rangle$. If $\{e\}^{\mathcal{L}}(a)\uparrow$ then $\langle e,a\rangle$ is the only i.s. of $\langle e_0,a\rangle$. If $\{e\}^{\mathcal{L}}(a) \simeq y$ then also $\langle e',y,a\rangle$ is an i.s. of $\langle e_0,a\rangle$.

v) If e_0 is not an index, or the length of a is not $(e_0)_2$ then $\langle e_0,a\rangle$ is the only i.s. of $\langle e_0,a\rangle$.

If $\{e_0\}^{\mathcal{L}}(a)\downarrow$ then this definition gives the same i.s. as the previous definition. The notion of a subcomputation can be defined as before. The subcomputations of $\langle e_0,a\rangle$ can be arranged as a tree. At each node in the tree there is a computation. $\langle e_0,a\rangle$ is put at the top node. If $\langle e',a'\rangle$ occurs at a node then the immediate subcomputations of $\langle e',a'\rangle$ occur at the nodes immediately below. This tree is called the computation tree of $\langle e,a\rangle$. At each node the branching has one of the following forms:



i) ii), ii') ii), ii'), iii), v) iv)

With these conventions the following is true:

<u>Theorem 1</u>: For all e,a : $\{e\}^{\mathcal{L}}(a)\downarrow$ iff the computation tree of $\langle e,a\rangle$ is wellfounded.

<u>Remark 1</u>: It turns out that clause XI (primitive recursion) is superfluous in the presence of clause XIII. It can be replaced by a primitive recursive function. When this function is added and

clause XI is omitted the same set of functions will be generated.

Computable functionals and fixpoint theorems:

Let $\mathcal{F}$ be a function: $X \to \omega$ where X is a subset of a cartesian product $P_{m_1} \times P_{m_2} \times \ldots \times P_{m_k} \times A^l$. P_m denotes the set of partial functions: $A^m \to S$. $\mathcal{F}$ is called a partial functional. $\mathcal{F}$ is monotone if $(\varphi_1 \ldots \varphi_k, a) \in \text{dom}\, \mathcal{F}$ and $\varphi_i \subseteq \psi_i$, $i = 1 \ldots k$ $\Rightarrow$ $(\psi_1 \ldots \psi_k, a) \in \text{dom}\, \mathcal{F}$ and $\mathcal{F}(\varphi_1 \ldots \varphi_k, a) \simeq \mathcal{F}(\psi_1 \ldots \psi_k, a)$. $\mathcal{F}$ is partial recursive in $\mathcal{L}$ if there is an index e such that for all $\varphi_1 \ldots \varphi_k, a$:

$$\mathcal{F}(\varphi_1 \ldots \varphi_k, a) \simeq \{e\}^{\mathcal{L}, \varphi_1 \ldots \varphi_k}(a) .$$

$\mathcal{F}$ is weakly partial recursive in $\mathcal{L}$ if there is a primitive recursive function $f(n_1 \ldots n_k)$ such that for all $e_1 \ldots e_k, a_1 \ldots a_k$, where a_i has length n_i, $i = 1 \ldots k$:

$$\mathcal{F}(\varphi_1 \ldots \varphi_k, a) \simeq \{f(n_1 \ldots n_k)\}^{\mathcal{L}}(e_1 \ldots e_k, a_1 \ldots a_k, a)$$

where $\varphi_i = \lambda b\, \{e_i\}^{\mathcal{L}}(b, a_i)$, $i = 1 \ldots k$.

First recursion theorem: Suppose that $\mathcal{F}$ is monotone and weakly partial recursive in $\mathcal{L}$, and that the domain of φ consists of tuples (φ, a) where the length of a is the same as the number of argument places in φ. Then there is a least φ such that for all a: $\mathcal{F}(\varphi, a) \simeq \varphi(a)$, and this φ is partial recursive in $\mathcal{L}$.

Second recursion theorem: $\forall e\, \exists x\, \forall a, \mathcal{L} : \{e\}^{\mathcal{L}}(x, a) \simeq \{x\}^{\mathcal{L}}(a)$.

Let Γ' be the inductive definition which is defined by the clauses I - XII in Γ. Then Γ' generates the class of primitive recursive functions. Let $\{e\}_{PRF}$ denote the primitive recursive function with index e.

Recursion theorem for primitive recursive functions:

Let $f(e,a)$ be primitive recursive. Then there is an e such that for all a: $f(e,a) = \{e\}_{PRF}(a)$.

Lemma 2: Let $\varphi = \lambda b\,\{e\}^{\mathcal{L}}(b,a)$. There is a primitive recursive function f such that for all x, c, y:

$$\{x\}^{\mathcal{L},\varphi}(c) \simeq y \iff \{f(x)\}^{\mathcal{L}}(c,a) \simeq y.$$

An immediate corollary of lemma 2 is

Lemma 3: If $\mathcal{F}$ is partial recursive in $\mathcal{L}$ then $\mathcal{F}$ is weakly partial recursive in $\mathcal{L}$.

Proof of lemma 2: We define a primitive recursive function g by cases. There is one case for each clause in the definition of Γ. $\varphi = \lambda b\,\{e\}^{\mathcal{L}}(b,a)$. Let b have length k and a have length l.

I $\quad x = \langle 1,n+1\rangle$. Let $g(x,t) = \langle 1,n+l+1\rangle$.

Clauses II - IX are treated similarly.

X $\quad x = \langle 10,n,e,e'\rangle$. Let $g(x,t) = \langle 10,n+l,g(e,t),g(e',t)\rangle$.

XIII $\quad x = \langle 13,n+1\rangle$. There is a primitive recursive function h such that for all $t,r,d,\mathcal{L}$: $\{h(t)\}^{\mathcal{L}}(r,d) \simeq \{\{t\}_{PRF}(r)\}^{\mathcal{L}}(d)$. Let $g(x,t) = h(t)$.

XV $\quad$ (the clause for application of φ): $x = \langle 15,k+n,i\rangle$. There is a primitive recursive function s such that for all d of length n: $\{e\}^{\mathcal{L}}(b,a) \simeq \{s(e,n)\}^{\mathcal{L}}(b,d,a)$ (d is a list of dummy arguments). Let $g(x,t) = s(e,n)$, where n = the length of c minus k.

By the recursion theorem for primitive recursive functions there is a t such that for all x: $g(x,t) = \{t\}_{PRF}(x)$. Let $f(x) = g(x,t)$ for this t. By induction on the length $|\{x\}^{\mathcal{L},\varphi}(c)|^{\mathcal{L},\varphi}$ one can prove: $\{x\}^{\mathcal{L},\varphi}(c) \simeq y \Rightarrow \{f(x)\}^{\mathcal{L}}(c) \simeq y$. By induction on $|\{f(x)\}^{\mathcal{L}}(c)|^{\mathcal{L}}$: $\{f(x)\}^{\mathcal{L}}(c) \simeq y \Rightarrow \{x\}^{\mathcal{L},\varphi}(c) \simeq y$.

$\square$

<u>Remark</u>: The converse of lemma 3 is not true. There are functionals which are weakly partial recursive in $\mathcal{L}$, and which are not partial recursive in $\mathcal{L}$. This can be proved by a cardinality argument as follows: Let

$T_1 = \{\varphi : \exists e,a \quad \varphi = \lambda x \{e\}^{\mathcal{L}}(x,a)\}$.

$T_2 = \{\varphi : \varphi$ is a unary partial function $A \to S\}$.

$T_3 = T_2 - T_1$. The cardinality of T_1, denoted by $\bar{\bar{T}}_1$, is $\bar{\bar{A}}$.

$\bar{\bar{T}}_2 = 2^{\bar{\bar{A}}}$. $\bar{\bar{T}}_3 = 2^{\bar{\bar{A}}}$.

Let $T_4 = \{\mathcal{F} : \exists e,a \ \mathcal{F} = \lambda\varphi \{e\}_{\mathcal{L},\varphi}(a)\}$, where φ ranges over T_2. $\bar{\bar{T}}_4 = \bar{\bar{A}}$. Let $T_5 = \{\mathcal{F} : \operatorname{dom}\mathcal{F} \subseteq T_3\}$. $\bar{\bar{T}}_5 = 2^{(2^{\bar{\bar{A}}})}$. Let $T_6 = \{\mathcal{F} : T_1 \subseteq \operatorname{dom}\mathcal{F}$ and $\varphi \in T_1 \Rightarrow \mathcal{F}(\varphi) = 0\}$. $\bar{\bar{T}}_6 = 2^{(2^{\bar{\bar{A}}})}$. Let $\mathcal{F} \in T_6 - T_4$. Then $\mathcal{F}$ is not partial recursive in $\mathcal{L}$, but obviously $\mathcal{F}$ is weakly partial rec. in $\mathcal{L}$.

§ 3 CONNECTION WITH KLEENE RECURSION IN HIGHER TYPES

Recursion in the present setting generalizes recursion in higher types as defined by Kleene. This can be seen as follows.

Let $n > 0$, and let $\epsilon_1 \dots \epsilon_k$ be a list of objects of type $n+1$, $F_1 \dots F_l$ a list of objects of type $n+2$. Let $\mathcal{K}$ denote the set of partial functions φ such that φ is recursive in $\epsilon_1 \dots \epsilon_k, F_1 \dots F_l$ in the sense of Kleene, and the domain of φ is a subset of a cartesion product $U_1 \times \dots \times U_m$, where $U_i = Tp(j)$ for some $j \leq n$ $(i = 1, \dots, m)$.

Let $S = Tp(0) \cup \dots \cup Tp(n-1)$. Let M be a primitive recursive (in the sense of Kleene) pairing function on ω $(= Tp(0))$ such that for all m,n: $M(n,m) > \max(m,n)$. Let K and L be the inverse functions of M. It is possible to extend M, K, L to S in such a way that

i) If $x \in Tp(i)$ $(i < n)$, $y \in Tp(j)$ $(j < n)$ then $M(x,y) \in Tp(k)$, where $k = \max(i,j)$.

ii) For each pair (i,j) such that $i < n$, $j < n$, $\sup(i,j) > 0$, the function f_{ij} is in $\mathcal{K}$, where f_{ij} is defined by: $f_{ij}(x,y,z) = M(x,y)(z)$, $x \in Tp(i)$, $y \in Tp(j)$, $z \in Tp(k-1)$, $k = \max(i,j)$.

iii) For each pair (i,j) such that $i < n$, $j < n$, $j \leq i$ the function $g_{ij} \in \mathcal{K}$, where $g_{ij}(x,y) = K(x)(y)$ if $K(x) \in Tp(j)$, $= 0$ if $K(x) \notin Tp(j)$, for $x \in Tp(i)$, $y \in Tp(j-1)$, $j > 0$. $g_{io}(x,y) = K(x)$ if $K(x) \in Tp(0)$, $= 0$ if $K(x) \notin Tp(0)$.

iv) Similar conditions for L.

Let $\mathcal{S} = (N, +1, M, K, L)$, $A = S \cup {}^{S}\omega$. ${}^{S}\omega$ can be regarded as the product $Tp(1) \times \dots \times Tp(n)$ since $S = Tp(0) \cup \dots \cup Tp(n-1)$. Hence

$Tp(n)$ can be regarded as a subset of S_ω via the natural injection $Tp(n) \longrightarrow \{0_1\} \times \{0_2\} \times \dots \times \{0_{n-1}\} \times Tp(n)$, where $0_i \in Tp(i)$ is the constant function with value 0.

We want to make a list $\mathcal{L}$ of functions and functionals such that $\mathcal{P}(\mathcal{L})$ is similar to $\mathcal{K}$. For trivial reasons $\mathcal{P}(\mathcal{L})$ cannot be equal to $\mathcal{K}$. For if $\varphi \in \mathcal{K}$ then

i) the domain of φ is a subset of a fixed cartesian product of types,

ii) the values of φ are in ω.

In $\mathcal{P}(\mathcal{L})$ there are functions which do not satisfy i). If $n > 1$ there are also functions which do not satisfy ii). But this difference between $\mathcal{K}$ and $\mathcal{P}(\mathcal{L})$ is not essential.

Let $\mathcal{L}$ be the list $g_1, g_2, \varepsilon_1' \dots \varepsilon_k'$, $F_1' \dots F_l', G, G_1 \dots G_{n-1}$ where

$$g_1(x) = \begin{cases} Tp(x) & \text{if } x \in S \\ n & \text{" } x \in S_\omega \end{cases}$$

$$g_2(x,y) = \begin{cases} x(y) & \text{if } x \in Tp(1),\ y \in Tp(0) \\ 0 & \text{otherwise} \end{cases}$$

$$\varepsilon_i'(x) = \begin{cases} \varepsilon_i\ (x \mid Tp(n-1) & \text{if } x \in S_\omega \\ 0 & \text{otherwise} \end{cases}$$

$$i = 1 \dots k$$

$$G(x,f) = \begin{cases} x(f' \mid Tp(i-2)) & \text{if } x \in Tp(i),\ 2 \le i < n \\ 0 & \text{otherwise} \end{cases}$$

$$G_i(f) = f' \mid Tp(i-1) \quad \text{for } 1 \le i < n$$

where $f'(x) = f(x)$ if $f(x) \in \omega$, $= 0$ otherwise,

$$F_i'(f) = F_i\ (f \mid Tp(n)),\ i = 1, \dots, l.$$

<u>Lemma 4</u>: There is a primitive recursive function $f(e)$ such that $\{e\}(a) \simeq m$ (in the sense of Kleene) iff $\{f(e)\}^{\mathcal{L}}(a') \simeq m$. (The list a can contain objects of type $\leq n$, and the objects $\epsilon_1 \dots \epsilon_k, F_1 \dots F_l$. a' is obtained by removing $\epsilon_1 \dots \epsilon_k, F_1 \dots F_l$.)

<u>Corollary</u>: $\mathcal{K} \subseteq \mathcal{P}(\mathcal{L})$.

<u>Remark</u>: Suppose $\varphi \in \mathcal{K}$. As a member of $\mathcal{K}$ the domain of φ is a subset of a cartesian product $Tp(i_1) \times \dots \times Tp(i_m)$ $(i_j \leq n)$. As a member of $\mathcal{P}(\mathcal{L})$ the domain of φ is a subset of A^m.

Suppose φ is a partial function such that the domain of φ is a subset of A^m, and the values of φ are in A. φ can be split into components in two ways. First we regard A as $S \cup {}^S\omega$. φ is split into φ' and φ'', where $\varphi'(a) \simeq \varphi(a)$ if $\varphi(a) \in S$, $\simeq 0$ if $\varphi(a) \in {}^S\omega$, undefined if $\varphi(a)$ is undefined. $\varphi''(a) \simeq \varphi(a)$ if $\varphi(a) \in {}^S\omega$, $\simeq 0$ if $\varphi(a) \in S$, undefined if $\varphi(a)$ is undefined, where $0 \in {}^S\omega$ is defined by: $0(r) = 0$ for all $r \in S$. φ'' is partial recursive (primitive recursive) in a list $\mathcal{L}$ if φ''' is, where $\varphi'''(a,y) \simeq \varphi''(a)(y)$ for all $y \in S$. <u>φ is partial recursive (primitive recursive) in $\mathcal{L}$</u> if φ' and φ'' are.

The other way of splitting up φ is natural when we regard A as $Tp(0) \cup Tp(1) \cup \dots \cup Tp(n-1) \cup (Tp(1) \times Tp(2) \times \dots \times Tp(n))$. Let $U = X_1 \times \dots \times X_m$ (φ is m-ary), where X_i is either $Tp(j)$ for some $j < n$, or X_i is $Tp(1) \times \dots \times Tp(n)$. Then $U \subseteq A^m$. U can be chosen in $(n+1)^m$ different ways. φ can be split into $(n+1)^m$ components, one for eacy U. Let φ_U be the restriction of φ to U. Each φ_U can be split into φ_{Ui}, $i \leq n$, where $\varphi_{Ui} : U \to Tp(i)$ if $i < n$, $\varphi_{Un} : U \to Tp(1) \times \dots \times Tp(n)$. If $i < n$ then φ_{Ui} is defined by:

$$\varphi_{Ui}(a) \simeq \begin{cases} \varphi(a) & \text{if } \varphi(a) \in Tp(i) \\ 0_i & \text{if } \varphi(a)\downarrow \text{ and } \varphi(a) \notin Tp(i) \\ \uparrow & \text{if } \varphi(a)\uparrow \end{cases}$$

where $0_i \in Tp(i)$ is defined by: $0_i(x) = 0$ for all $x \in Tp(i-1)$ if $i > 0$. φ_{Un} is defined as φ_{Ui} with $Tp(i)$ replaced by S_ω. φ_{Un} can be split into φ_{Unj}, $1 \leq j \leq n$, where $\varphi_{Unj}(a)$ is the j-th component of $\varphi_{Un}(a)$. Hence $\varphi_{Unj} : U \to Tp(j)$. A partial function $\psi : U \to Tp(i)$, $i > 0$ is partial recursive (primitive recursive) in the sense of Kleene if ψ' is, where $\psi' : U \times Tp(i-1) \to Tp(0)$ is defined by: $\psi'(x,y) \simeq \psi(x)(y)$. <u>$\varphi$ is partial recursive (primitive recursive) in the sense of Kleene</u> if all these components are.

<u>Lemma 5</u>: Suppose $\varphi \in \mathcal{P}(\mathcal{L})$. Then φ is partial recursive in $\varepsilon_1 \ldots \varepsilon_k, F_1 \ldots F_l$ in the sense of Kleene.

<u>Corollary</u>: Let R be a subset of $Tp(n)$. Then R is recursive (recursively enumerable) in $\varepsilon_1 \ldots \varepsilon_k, F_1 \ldots F_l$ in the sense of Kleene iff R is recursive (recursively enumerable) in $\mathcal{L}$.

<u>Lemma 6</u>: Let $f : A^n \to A$. Then f is primitive recursive in $g_1, g_2, G, G_1, \ldots G_{n-1}$ iff f is primitive recursive in the sense of Kleene.

Hence the following definition is meaningful: f is <u>primitive recursive</u> if f is primitive recursive in the sense of Kleene. This definition will be used in the rest of this chapter even if it is not the same as the one given in § 2.

Let $\mathcal{L}$ be the list $g_1, g_2, G, G_1 \ldots G_{n-1}, f_1 \ldots f_k, F_1 \ldots F_l$ where f_i is a total function $A^n \to S$ for $1 \leq i \leq k$, F_i is a total functional with values in S. We want to find a list of objects of type $n+1$ and $n+2$ such that recursion in these objects (in the sense of Kleene) is essentially the same as recursion in $\mathcal{L}$. To construct this list we need two primitive recursive functions p and q between A and $Tp(n)$ such that $p : A \to Tp(n)$, ($Tp(n)$ is regarded as a subset of A), $q : A \to A$, $q(p(x)) = x$ for all x in A. p and q can be constructed from the functions $\langle\langle\ \rangle\rangle$, p_i^j, q_i^j, where $\langle\langle\ \rangle\rangle : \bigcup_{k<\omega} Tp(n)^k \xrightarrow[\text{one-one}]{} Tp(n)$, $p_i^j : Tp(i) \to Tp(j)$, $q_i^j : Tp(j) \to Tp(i)$ for $i < j \leq n$, $q_i^j(p_i^j(\alpha^i)) = \alpha^i$ for all $\alpha \in Tp(i)$. Descriptions of $\langle\langle\ \rangle\rangle$, p_i^j, q_i^j can be found in the works of Kleene [12]. For each k the restriction of $\langle\langle\ \rangle\rangle$ to $Tp(n)^k$ is primitive recursive. So are the functions p_i^j, q_i^j, $\lambda x\, i(x)_i$, lh where $(\langle\langle\alpha_1^n \ldots \alpha_n^n\rangle\rangle)_i = \alpha_i^n$, $lh(\langle\langle\alpha_1^n \ldots \alpha_m^n\rangle\rangle) = m$.

Definition of p: $p(r) = \langle\langle \underline{i}, p_i^n(r)\rangle\rangle$ if $r \in Tp(i)$, $i<n$. $\underline{i} \in Tp(n)$ denotes the constant function with value i. If $x \in {}^{S}\omega = Tp(1) \times \ldots \times Tp(n)$ then $x = (\alpha^1 \ldots \alpha^n)$ where $\alpha^i \in Tp(i)$, $i = 1,\ldots n$. Let $p(x) = \langle\langle \underline{n}, p_1^n(\alpha^1), p_2^n(\alpha^2), \ldots, p_{n-1}^n(\alpha^{n-1}), \alpha^n\rangle\rangle$. Let q be defined by: $q(r) = 0$ if $r \in S$. If $x \in {}^{S}\omega$, $x = (\alpha^1, \alpha^2, \ldots, \alpha^n)$ let

$$
\begin{aligned}
q(x) &= q_i^n(\alpha^n)_2 \quad \text{if } lh(\alpha^n) = 2 \text{ and } (\alpha^n)_1(0_{n-1}) = i, \\
&= (q_1^n((\alpha^n)_2), q_2^n((\alpha^n)_3), \ldots, q_{n-1}^n((\alpha^n)_n), (\alpha^n)_{n+1}) \\
&\qquad \text{if } lh(\alpha^n) = n+1 \text{ and } (\alpha^n)_1(0_{n-1}) = n \\
&= 0 \quad \text{otherwise.}
\end{aligned}
$$

It is routine work to prove that p and q are primitive recursive, and that $q(p(x)) = x$ for all x.

Let $f : A^m \to S$ be one of the functions in the list $\mathcal{L}$. Let $f' \in Tp(n+1)$ be defined by

$$f'(\alpha^n) = [pf(q((\alpha^n)_{1,1}), q((\alpha^n)_{1,2}), \ldots, q((\alpha^n)_{1,m}))](q_{n-1}^n((\alpha^n)_2)),$$

where $(\alpha^n)_{1,i} = ((\alpha^n)_1)_i$. Then all information about f is contained in f'. This can be seen as follows. Suppose $f(x_1 \ldots x_m) = r$. Let $\gamma^n = \langle\langle p(x_1), p(x_2), \ldots, p(x_m)\rangle\rangle$, $\alpha^n = \langle\langle \gamma, p^n_{n-1}\beta\rangle\rangle$ for some $\beta \in Tp(n-1)$. Then $f'(\alpha^n) = [pf(x_1, \ldots, x_m)](\beta) = [p(r)](\beta)$. Hence $p(r) = \lambda\beta^{n-1} f'(\alpha^n)$, and $r = q(\lambda\beta^{n-1} f'(\alpha^n))$.

Let F be one of the functionals in the list $\mathcal{L}$. Let $F' \in Tp(n+2)$ be defined by the following description: Let $\alpha \in Tp(n+1)$. Split α into $(\alpha)_1 = \beta$ and $(\alpha)_2 = \gamma$. Let f be the function from A to S defined by: $f(x) = q\, q^{n+1}_n\, \beta' p(x)$ if $q\, q^{n+1}_n\, \beta' p(x) \in S$; $f(x) = 0$ if $q\, q^{n+1}_n \beta' p(x) \in S_\omega$.
β' is the function from $Tp(n)$ to $Tp(n+1)$ defined by: $\beta'(\delta^n) = \lambda\varepsilon^n \beta(\langle\langle \varepsilon, \delta\rangle\rangle)$. As β varies through $Tp(n+1)$, all functions f from A to S will be generated in this way. Suppose $F(f) = y$. Let $F'(\alpha) = p(y)(q^{n+1}_{n-1}(\gamma))$. Then all information about F is contained in F'.

<u>Lemma 7</u>: There is a primitive recursive function $' : \omega \to \omega$ such that for all $e, x_1 \ldots x_m, y$:
$\{e\}^{\mathcal{L}}(x_1 \ldots x_m) \simeq y \iff \lambda\beta\{e'\}(p(x_1) \ldots p(x_m), \beta) = p(y)$, where β ranges over $Tp(n-1)$, $\{e'\}$ denotes the e'-th function which is partial recursive in $f'_1, \ldots, f'_k, F'_1, \ldots, F'_i$ in the sense of Kleene.

Proof: We define a primitive recursive function $g : \omega^2 \to \omega$. g is defined by cases. There is one case for each clause in the inductive definition Γ. By the recursion theorem for primitive recursive functions there is a number k such that $g(e,k) = \{k\}_{PRF}(e)$ for all e. Let $e' = g(e,k)$ for this k. It is explained below how to define $g(e,k)$ in the cases diagonalization, application of f_i, and application of F_j.

Diagonalization: $e = \langle 13, h+1\rangle$. If $\{e\}^{\mathcal{L}}(\hat{e},a) \simeq x$ then $\{\hat{e}\}^{\mathcal{L}}(a) \simeq x$. There is a primitive recursive function $l(k)$ such that for all $k,\hat{e},a,\beta$:

$$\{l(k)\}(p(\hat{e}),p(a),\beta) \simeq \{\{k\}_{PRF}(\hat{e})\}(p(a),\beta).$$

Let $g(e,k) = l(k)$.

Application of f_i : $e = \langle 15, m+h, i\rangle$. Let $g(e,k) = e_o$, where e_o is an index such that $\{e_o\}(p(x_1)\dots p(x_m),p(a),\beta) \simeq f'(\alpha)$, where $\alpha = \langle\langle\ \langle\langle p(x_1)\dots p(x_m)\rangle\rangle\ ,\ p^n_{n-1}(\beta)\rangle\rangle$.

Application of F_j : $e = \langle 16,h,\hat{e},j\rangle$. Let $g(e,k)$ be an index such that $\{g(e,k)\}(p(a),\beta) \simeq F'(\alpha)$ where $\alpha \in Tp(n+1)$ is defined such that $(\alpha)_1(\langle\langle\langle \epsilon,p(x)\rangle\rangle\rangle) = (p^{n+1}_n[\lambda\gamma^{n-1}\{g(\hat{e},k)\}(p(x),p(a),\gamma)])(\epsilon)$ and $(\alpha)_2 = p^{n+1}_{n-1}(\beta^{n-1})$.

By induction on the length of $\{e\}^{\mathcal{L}}(x_1\dots x_m)$ it can be proved that $\{e\}^{\mathcal{L}}(x_1\dots x_m) \simeq y \Rightarrow \lambda\beta\{e'\}(p(x_1)\dots p(x_m),\beta) = p(y)$. To prove the induction step for the case application of F_j, suppose $\{e\}^{\mathcal{L}}(a) \simeq F_j(\lambda x\{\hat{e}\}^{\mathcal{L}}(x,a)) \simeq y$. Then $|\{\hat{e}\}^{\mathcal{L}}(x,a)| < |\{e\}^{\mathcal{L}}(a)|$ for all x, and $\{\hat{e}\}^{\mathcal{L}}(x,a) \simeq y' \Rightarrow \lambda\gamma\{\hat{e}'\}(p(x),p(a),\gamma) = p(y')$. $\{e'\}(p(a),\beta) \simeq F'(\alpha)$ by the construction of $'$, where α is described above. As in the description of F' let f be defined by:

$$\begin{aligned} f(x) &= q\ q^{n+1}_n(\alpha)'_1\,(p(x)) \\ &= q\ q^{n+1}_n(\lambda\,\epsilon\,(\alpha)_1\,(\langle\langle\langle\epsilon,p(x)\rangle\rangle\rangle)) \\ &= q\ q^{n+1}_n(\lambda\,\epsilon\,(p^{n+1}_n[\lambda\gamma\{\hat{e}'\}(p(x),p(a),\gamma)])(\epsilon)) \\ &= q\ q^{n+1}_n\ p^{n+1}_n[\lambda\gamma\{\hat{e}'\}(p(x),p(a),\gamma)] \\ &= q\ [\lambda\gamma\{\hat{e}'\}(p(x),p(a),\gamma)] \\ &= q\ p(y') \quad \text{by induction hypothesis,} \end{aligned}$$

where $\{\hat{e}\}^{\mathcal{L}}(x,a) \simeq y'$. So $f(x) = y'$. Hence $f = \lambda x\{\hat{e}\}^{\mathcal{L}}(x,a)$.
By the description of F'

$$\begin{aligned} F'(\alpha) &= p(y)(q^{n+1}_{n-1}((\alpha)_2)) \\ &= p(y)(q^{n+1}_{n-1}\ p^{n+1}_{n-1}(\beta^{n-1})) \\ &= p(y)(\beta^{n-1}). \end{aligned}$$

Hence $\lambda\beta^{n-1}\{e'\}(p(a),\beta) = p(y)$.

By induction on $\min\{|\{e'\}(p(x_1)\ldots p(x_m),\beta)| : \beta \in Tp(n-1)\}$ it can be proved that for all $e', x_1 \ldots x_m, y$:

$$\lambda\beta\{e'\}(p(x_1)\ldots p(x_m),\beta) = p(y) \Rightarrow \{e\}^{\mathcal{L}}(x_1\ldots x_m) \simeq y$$

$\square$

In a similar way one can prove:

Lemma 8: There is a primitive recursive function $'': \omega \to \omega$ such that for all $e, \alpha_1 \ldots \alpha_m, h$:

$$\{e\}(\alpha_1\ldots\alpha_m) \simeq h \iff \{e''\}^{\mathcal{L}}(\alpha_1\ldots\alpha_m) \simeq h,$$

where α_i ranges over $Tp(j_i)$, $0 \leq j_i \leq n$, $i = 1\ldots m$.

The purpose of this paper is to reprove some results about recursion in higher types within the framework of chapters 1 and 2. The following should be true: Suppose we have proved a result about recursion on $\mathfrak{A}$. Then there is an easy way to deduce a similar result for recursion in higher types.

When $S = Tp(0) \cup \ldots \cup Tp(n-1)$ we have seen that there is a close correspondence between recursion on $\mathfrak{A}$ and recursion in the sense of Kleene. This correspondence will be utilized in the transition between the two kinds of recursion.

§ 4 RECURSION IN NORMAL LISTS ON $\mathfrak{A}$

In this chapter we will study recursion in lists $\mathcal{L}$ which are normal, i.e. the functional E defined below is weakly recursive in $\mathcal{L}$, the equality relation on S is recursive in $\mathcal{L}$, and $\mathcal{L}$ contains no partial functions.

$$E(f) = \begin{cases} 0 & \text{if } \exists x\ f(x) = 0 \\ 1 & \text{if } \forall x \exists y \neq 0 \quad f(x) = y \end{cases}$$

where f is a total function from A to S.

Remark 1: Suppose $S = Tp(0) \cup \ldots \cup Tp(n-1)$. Let $\mathcal{L}$ be a normal list. Construct the list of objects of type $n+1$ and $n+2$ as described in § 3. It can be proved that ${}^{n+2}E$ is weakly recursive in this list, where ${}^{n+2}E \in Tp(n+2)$ is defined by:

$${}^{n+2}E(\alpha^{n+1}) = \begin{cases} 0 & \text{if } \exists \beta \in Tp(n) \quad \alpha(\beta) = 0 \\ 1 & \text{if } \forall \beta \in Tp(n) \quad \alpha(\beta) \neq 0 \end{cases}$$

The opposite is also true. Given a list of objects of type $n+1$ and $n+2$, let $\mathcal{L}$ be the list constructed in § 3. If ${}^{n+2}E$ is weakly recursive in the objects of type $n+1$ and $n+2$, then $\mathcal{L}$ is normal.

In this case the statement "the equality relation on S is recursive in $\mathcal{L}$ " is superfluous in the definition of the notion "normal". The statement can be proved from the fact that E is weakly recursive in $\mathcal{L}$.

Remark 2: In works on higher types the notion "normal" is often defined in a stronger way than here: An object $R \in Tp(n+2)$ is normal if ${}^{n+2}E$ is recursive in F. Here "recursive" is replaced by "weakly recursive". This weaker notion is chosen because it is sufficient in many proofs. (In theorem 5 it is not sufficient.)

Remark 3: If $\mathcal{L}$ is normal then the relations which are recursive in $\mathcal{L}$ are closed under the quantifiers $\forall$ and $\exists$, i.e. if R is recursive in $\mathcal{L}$ then so are $\forall x\, R$ and $\exists x\, R$.

Let $\mathcal{L}$ be a list, and let $C^{\mathcal{L}} \subseteq A$ be defined by:

$$C^{\mathcal{L}} = \{\langle e,a\rangle : \{e\}^{\mathcal{L}}(a)\downarrow\}.$$

Theorem 2: Let $\mathcal{L}$ be normal. There is a function p which is partial recursive in $\mathcal{L}$ such that:

$$\begin{aligned} x \in C^{\mathcal{L}} \text{ or } y \in C^{\mathcal{L}} &\iff p(x,y)\downarrow, \\ x \in C^{\mathcal{L}} \text{ and } |x|^{\mathcal{L}} \leq |y|^{\mathcal{L}} &\Rightarrow p(x,y) = 0, \\ |x|^{\mathcal{L}} > |y|^{\mathcal{L}} &\Rightarrow p(x,y) = 1. \end{aligned}$$

Remark: The index of p can be found in a uniform way. It is a primitive recursice function of t, where t is an index for the primitive recursive function which proves that E is weakly recursive in $\mathcal{L}$.

Proof of theorem 2: Define the partial functional $\mathcal{F}$ be cases. There is one case for each pair of clauses in the inductive definition of Γ. The form of the sequences x and y tells which case we are in. Because there are so many cases (225) only one will be given here: when x corresponds to clause X (substitution) and y to clause XVI (application of F).

Suppose $x = \langle\langle 10,n,e,e'\rangle,a\rangle$ and $y = \langle\langle 16,m,t,1\rangle,b\rangle$. Let p be a partial function, and let

$$\varphi_1(x,y) \simeq E(\lambda z\ p(\langle e,a\rangle,\langle t,z,b\rangle))$$

$$\varphi_2(x,y) \simeq E(\lambda z\ p(\langle e',\{e\}^{\mathscr{L}}(a),a\rangle,\langle t,z,b\rangle))$$

$$f(x,y) = \begin{cases} 0 & \text{if } x=0 \text{ and } y=0 \\ 1 & \text{if } x\neq 0 \text{ or } y\neq 0 \end{cases}$$

Let $\mathscr{F}(p,x,y) \simeq f(\varphi_1(x,y),\varphi_2(x,y))$.

Then φ_1, φ_2 are partial recursive in $\mathscr{L}$, p (since $\mathscr{L}$ is normal), and f is primitive recursive. $\mathscr{F}$ is partial recursive in F, and monotone. If $\lambda z\ p(\langle e,a\rangle,\langle t,z,b\rangle)$ and $\lambda z\ p(\langle e',\{e\}^{\mathscr{L}}(a),a\rangle,\langle t,z,b\rangle)$ are total then $\varphi_1(x,y)\downarrow$ and $\varphi_2(x,y)\downarrow$, and $\mathscr{F}(p,x,y)\downarrow$. Also

$$\begin{aligned}\mathscr{F}(p,x,y) \simeq 0 \quad & \text{if } \exists z\ p(\langle e,a\rangle,\langle t,z,b\rangle) \simeq 0 \\ & \text{and } \exists z\ p(\langle e',\{e\}^{\mathscr{L}}(a),a\rangle,\langle t,z,b\rangle) \simeq 0 \\ \simeq 1 \quad & \text{if } \forall z\ \exists v\neq 0\ p(\langle e,a\rangle,\langle t,z,b\rangle) \simeq v \\ & \text{or } \forall z\ \exists v\neq 0\ p(\langle e',\{e\}^{\mathscr{L}}(a),a\rangle,\langle t,z,b\rangle) \simeq v\end{aligned}$$

Let p be a solution to the equality $\forall xy(\mathscr{F}(p,x,y) \simeq p(x,y))$. By induction on $\min\{|x|^{\mathscr{L}},|y|^{\mathscr{L}}\}$ one can prove:

$$x \in C^{\mathscr{L}} \text{ or } y \in C^{\mathscr{L}} \Rightarrow p(x,y)\downarrow,$$

$$x \in C^{\mathscr{L}} \text{ and } |x|^{\mathscr{L}} \leq |y|^{\mathscr{L}} \Rightarrow p(x,y) = 0,$$

$$|x|^{\mathscr{L}} > |y|^{\mathscr{L}} \Rightarrow p(x,y) = 1.$$

The induction goes as follows: Suppose x and y are as in the case above. If $x \in C^{\mathscr{L}}$ then the immediate subcomputations of x are $\{e\}^{\mathscr{L}}(a)$ and $\{e'\}^{\mathscr{L}}(\{e\}^{\mathscr{L}}(a),a)$, and $|\{e\}^{\mathscr{L}}(a)| < |x|^{\mathscr{L}}$, $|\{e'\}^{\mathscr{L}}(\{e\}^{\mathscr{L}}(a),a)| < |x|$. If $y \in C^{\mathscr{L}}$ then $\{t\}^{\mathscr{L}}(z,b)$ is an immediate subcomputation of y for all $z \in A$, and $|\{t\}^{\mathscr{L}}(z,b)| < |y|$ for all z. Suppose $x \in C^{\mathscr{L}}$ and $|x| \leq |y|$. Then by induction hypothesis $p(\langle e,a\rangle,\langle t,z,b\rangle)\downarrow$ and $p(\langle e',\{e\}^{\mathcal{L}}(a),a\rangle,\langle t,z,b\rangle)\downarrow$ for all z. Also $\exists z\ |\{e\}^{\mathcal{L}}(a)| \leq |\{t\}^{\mathcal{L}}(z,b)|$ and

$$\exists z\ |\{e'\}^{\mathcal{L}}(\{e\}^{\mathcal{L}}(a),a)| \leq |\{t\}^{\mathcal{L}}(z,b)| .$$

Hence $\exists z\ p(\langle e,a\rangle,\langle z,t,b\rangle) \simeq 0$ and
$\exists z\ p(\langle e',\{e\}^{\mathcal{L}}(a),a\rangle,\langle t,z,b\rangle) \simeq 0$.
Hence $\mathcal{F}(p,x,y) \simeq 0$, and $p(x,y) \simeq 0$ since p is a solution to the equality. A similar argument applies when $|x| > |y|$. This proves the induction.

Let p be the function defined in the theorem. By looking at the definition of $\mathcal{F}$ it can be seen that p is a solution to the equality. By the above induction p is the least solution. By the first recursion theorem p is partial recursive in F. $\square$

Now we can prove the existence of selection operators for natural numbers.

<u>Theorem 3</u>: Suppose $\mathcal{L}$ is normal. Then there is a function φ which is partial recursive in $\mathcal{L}$ such that for all e,a: If $\exists n \in N\ \ \{e\}^{\mathcal{L}}(n,a)\downarrow$ then $\varphi(e,a)\downarrow$, and $\{e\}^{\mathcal{L}}(\varphi(e,a),a)\downarrow$. Moreover if $\varphi(e,a) \simeq n$ then $\{e\}^{\mathcal{L}}(n,a)\downarrow$. The index of φ is a primitive recursive function of the number of places in a.

<u>Proof</u>: Let ψ be defined by:

$$\begin{aligned}\psi(r,n,e,a) &\simeq n \quad \text{if } |\{e\}^{\mathcal{L}}(n,a)| \leq |\{r\}^{\mathcal{L}}(n+1,e,a)| \\ &\simeq \{r\}^{\mathcal{L}}(n+1,e,a) \quad \text{if } |\{e\}^{\mathcal{L}}(n,a)| > |\{r\}^{\mathcal{L}}(n+1,e,a)| .\end{aligned}$$

Choose r such that $\psi(r,n,e,a) \simeq \{r\}^{\mathcal{L}}(n,e,a)$ for all n,e,a.
Let $\varphi(e,a) \simeq \{r\}^{\mathcal{L}}(0,e,a)$.

I : $\{e\}(n,a)\downarrow \Rightarrow \{r\}^{\mathcal{L}}(n,e,a)\downarrow$,

II : $\{r\}^{\mathcal{L}}(n+1,e,a)\downarrow \Rightarrow \{r\}^{\mathcal{L}}(n,e,a)\downarrow$.

From I and II it follows:

$$\exists n\{e\}^{\mathcal{L}}(n,a)\downarrow \Rightarrow \{r\}^{\mathcal{L}}(0,e,a)\downarrow .$$

Suppose $\{r\}^{\mathcal{L}}(0,e,a) \simeq k$. Want to prove that $\{e\}(k,a)\downarrow$. There is an n such that $|\{e\}(n,a)| \leq |\{r\}(n+1,e,a)|$, for in the opposite case $\{r\}^{\mathcal{L}}(n,e,a) \simeq \{r\}^{\mathcal{L}}(n+1,e,a) \simeq k$ for all n, and $|\{r\}^{\mathcal{L}}(n,e,a)| > |\{r\}^{\mathcal{L}}(n+1,e,a)|$ for all n. Hence we have obtained an infinite descending sequence of ordinals, a contradiction. Let n be the least m such that $|\{e\}^{\mathcal{L}}(m,a)| \leq |\{r\}^{\mathcal{L}}(m+1,e,a)|$. Then $\{e\}^{\mathcal{L}}(n,a)\downarrow$, and $\{r\}^{\mathcal{L}}(n,e,a) \simeq n$. For all $n' < n$ $\{r\}^{\mathcal{L}}(n',e,a) \simeq \{r\}^{\mathcal{L}}(n'+1,e,a) \simeq n$. Hence $\{r\}^{\mathcal{L}}(0,e,a) \simeq n$, i.e. $n = k$, and $\{e\}^{\mathcal{L}}(k,a)\downarrow$. □

Corollary: If the relations R_1, R_2 are recursively enumerable in $\mathcal{L}$, then so are $\exists n\, R_1$, $R_1 \vee R_2$.

Lemma 9: Suppose that the equality relation on S is recursive in $\mathcal{L}$, and that the functional E_S defined in lemma 1 is weakly recursive in $\mathcal{L}$. Then there is a relation $S(x,y)$ which is recursively enumerable in $\mathcal{L}$ such that if $x \in C^{\mathcal{L}}$ (= the set of convergent computations) then $S(x,y) \Leftrightarrow y$ is an immediate subcomputation of x. The set $\{y: S(x,y)\}$ is recursive in $x, \mathcal{L}$ when $x \in C^{\mathcal{L}}$.

Proof: $S(x,y)$ is defined by cases. There is one case for each clause in Γ. The form of x determines which case is to be applied. We need the functions lh, $\lambda x\, i(x)_i$, and the predicate Seq to decide the form of x; the graphs of these functions and the predicate are primitive recursive in the equality relation on S and the functional E_S (by lemma 1). The only interesting case is when x is a substitution. In the other cases it can be recursively decided whether or not $S(x,y)$ is satisfied. So let $x = \langle\langle 10,n,e,e'\rangle,a\rangle$. Then

$$S(x,y) \iff y = \langle e,a\rangle \text{ or } (\{e\}^{\mathcal{L}}(a)\downarrow \text{ and } y = \langle e',\{e\}^{\mathcal{L}}(a),a\rangle).$$

The relation on the right side is recursively enumerable in $\mathcal{L}$. If $x \in C^{\mathcal{L}}$ then $\{e\}^{\mathcal{L}}(a)\downarrow$, and the relation is recursive in x, $\mathcal{L}$.

<u>Lemma 10</u>: Suppose $\mathcal{L}$ is normal. If $\{e\}^{\mathcal{L}}(a)\downarrow$ then the computation tree of $\{e\}^{\mathcal{L}}(a)$ is recursive in $\mathcal{L}$, a.

<u>Proof</u>: Let q be the partial function defined by:

$q(x,y)\downarrow$ iff $x \in C^{\mathcal{L}}$,

y is a subcomputation of $x \implies q(x,y) = 0$,

y is not a subcomputation of $x \implies q(x,y) = 1$.

Then q is a fixpoint for the monotone $\mathcal{L}$-recursive functional $\mathcal{F}$ defined by:

$$\begin{aligned} \mathcal{F}(q,x,y) &\simeq 0 \quad \text{if } x \in C^{\mathcal{L}} \text{ and } S(x,y) \\ &\qquad\qquad \text{or } \exists z(S(x,z) \text{ and } q(z,y) \simeq 0), \\ &\simeq 1 \quad \text{if } x \in C^{\mathcal{L}} \text{ and } \neg S(x,y) \\ &\qquad\qquad \text{and } \forall z(S(x,z) \implies q(z,y) \simeq 1). \end{aligned}$$

The quantifiers $\exists$ and $\forall$ can be expressed by E. Hence $\mathcal{F}$ is recursive in $\mathcal{L}$. Suppose q' is a fixpoint for $\mathcal{F}$. By induction on $|x|^{\mathcal{L}}$ it can be proved that $q(x,y)\downarrow \implies q(x,y) \simeq q'(x,y)$. Hence q' is an extension of q. Hence q is the least fixpoint of $\mathcal{F}$, and by the first recursion theorem q is partial recursive in $\mathcal{L}$.

If $\{e\}^{\mathcal{L}}(a)\downarrow$, then $\lambda y\, q(\langle e,a\rangle,y)$ is the characteristic function of the computation tree of $\{e\}^{\mathcal{L}}(a)$. $\square$

Let Y be a set of elements in ${}^{S}\omega$, indexed by S, i.e. $Y = \{\alpha_r : r \in S\}$. Then all elements in Y can be coded by one element in ${}^{S}\omega$, namely α defined by:

$$\alpha(r) = \alpha_{(r)_1}((r)_2).$$

For all $r \in S$ $\lambda s\, \alpha(\langle r,s\rangle) = \alpha_r$. This property will be utilized in the next theorem.

There is a one-one function ** from A into ${}^{S}\omega$ and a function $^{--} : A \to A$ such that the graphs of ** and $^{--}$ are primitive recursive in the equality relation on S and the functional E_S, and $(x^{**})^{--} = x$ for all x. ** and $^{--}$ can for instance be defined by:

$$r^{**} = \langle r^*,\underline{0}\rangle$$

$$\alpha^{**} = \langle \alpha,\underline{1}\rangle$$

where $\underline{0} \in {}^{S}\omega$ and $\underline{1} \in {}^{S}\omega$ denote the constant functions with values 0 and 1 respectively.

$$x^{--} = \begin{cases} (x)_1 & \text{if } (x)_2(0) = 1 \\ (x)_1^{-} & \text{if } (x)_2(0) = 0. \end{cases}$$

<u>Theorem 4</u>: Suppose that the equality relation on S is recursive in $\mathcal{L}$, and that E_S is weakly recursive in $\mathcal{L}$. Then there is a relation R which is recursively enumerable in $\mathcal{L}$ such that for all e,a: $\{e\}^{\mathcal{L}}(a)\uparrow \iff \exists\alpha\, R(\alpha,\langle e,a\rangle)$.

Proof: $\{e\}^{\mathscr{L}}(a)\uparrow \iff$ the computation tree of $\{e\}^{\mathscr{L}}(a)$ is not well-founded $\iff$

$\exists \alpha_0\, \alpha_1 \ldots \alpha_n \ldots [\bar{\bar{\alpha_0}} = \langle e,a\rangle$ and

$\forall i(\bar{\bar{\alpha_{i+1}}}$ is an immediate subcomputation of $\bar{\bar{\alpha_i}})]$

$\iff \exists \alpha_0\, \alpha_1 \ldots \alpha_n \ldots (\bar{\bar{\alpha_0}} = \langle e,a\rangle$ and $\forall i\, S(\bar{\bar{\alpha_i}},\bar{\bar{\alpha_{i+1}}}))$

$\iff \exists \alpha(\overline{\overline{\alpha[0]}} = \langle e,a\rangle$ and $\forall i\, S(\overline{\overline{\alpha[i]}},\overline{\overline{\alpha[i+1]}}))$

where $\alpha[r] = \lambda s\, \alpha(\langle r,s\rangle)$. Let $R(\alpha,\langle e,a\rangle)$

$\iff \overline{\overline{\alpha[0]}} = \langle e,a\rangle$ and $\forall i\, S(\overline{\overline{\alpha[i]}},\overline{\overline{\alpha[i+1]}})$.

Then R is recursively enumerable in $\mathscr{L}$. □

Corollary: The relations which are recursively enumerable in $\mathscr{L}$ are not closed under existential quantifiers ranging over S_ω.

Let $P(A)$ denote the set of subsets of A. A relation $R \subseteq P(A) \times A$ is recursive in $\mathscr{L}$ if there is an index e such that $\lambda X\, x\{e\}^{\mathscr{L},X}(x)$ is the characteristic function of R.

Theorem 5: Suppose that $\mathscr{L}$ is normal and that the functional E is recursive in $\mathscr{L}$. Let $B \subseteq A$. Then B is recursively enumerable in $\mathscr{L}$ iff there is a relation $R \subseteq P(A) \times A$ which is recursive in $\mathscr{L}$ such that for all $x \in A$: $x \in B \iff \exists X$ (X is recursive in $x,\mathscr{L}$ and $(X,x) \in R$).

Proof: Suppose B is recursively enumerable in $\mathscr{L}$. Let e_0 be an index such that for all x: $x \in B \iff \{e_0\}^{\mathscr{L}}(x) \simeq 0$.

We define a relation $S_X(x,y)$ by cases on x. There is one case for each of the clauses in the definition of Γ. Some of the cases are given below: If x is a starting computation then

$\neg S_X(x,y)$ for all y. If $x = \langle\langle 10,n,e,e'\rangle,a,z\rangle$ (composition) then $S_X(x,y) \iff x \in X$ and $y \in X$ and $\exists u(y = \langle e,a,u\rangle$ or $y = \langle e',u,a,z\rangle)$. If $x = \langle\langle 16,n,e,i\rangle,a,z\rangle$ (application of the functional F_i) then $S_X(x,y) \iff x \in X$ and $y \in X$ and $\exists x'y'\ y = \langle e,x',a,y'\rangle$.

If x is not of a form which corresponds to a clause in Γ then $S_X(x,y) \iff x \in X$ and $y \in X$ and $x = y$. As a relation of X, x,y S is recursive in $\mathcal{L}$, and $S_X(x,y)$ says that y is an immediate subcomputation of x with respect to X.

Let R be defined by:

$$R(X,x) \iff \langle e_0,x,0\rangle \in X \text{ and}$$

$$\forall n\,a\,y\,y'\,(\langle n,a,y\rangle \in X \text{ and } \langle n,a,y'\rangle \in X \Rightarrow y = y')$$

$$\text{and } \forall x(x \in X \Rightarrow \mathrm{Seq}(x) \wedge \mathrm{lh}(x) \geq 2)$$

$$\text{and } \forall\alpha\,\exists i \quad S_X\,(\alpha[i]^{--},\alpha[i+1]^{--})$$

$$\text{and } Q(X)$$

where Q is the relation which says that if $x \in X$ then x is a convergent computation, and for all y: y is an immediate subcomputation of x iff $S_X(x,y)$.

Obviously all parts of the definition of R except Q are recursive in $\mathcal{L}$. To prove that R is recursive in $\mathcal{L}$ we give instructions how to compute the characteristic function of R. First see if all parts of $R(X,x)$ except $Q(X)$ are satisfied. If not give output 1. If these conditions are satisfied then X can be arranged as a wellfounded tree. Let t_X be the function defined by:

$$t_X(x) \simeq 1 \quad \text{if } x \notin X$$

$$\simeq 1 \quad \text{if } x \in X \text{ and } t_X(y) \simeq 1 \text{ for some } y \text{ below } x \text{ in the tree.}$$

In addition there is a case for each clause in the definition of Γ. The cases which correspond to a starting computation, composition and application of F_i are given below:

Starting computation: $t_X(x) \simeq 0$ if $x = \langle n,a,y\rangle$ and $\langle n,a,y\rangle$ is a starting computation and $\{n\}^{\mathcal{L}}(a) \simeq y$. $t_X(x) \simeq 1$ if $\exists y'\ \{n\}^{\mathcal{L}}(a) \simeq y'$ and $y' \neq y$.

Composition: $x = \langle\langle 10,n,e,e'\rangle,a,z\rangle$.

$$\begin{aligned} t_X(x) &\simeq 0 \quad \text{if } \exists u[t_X(\langle e,a,u\rangle) \simeq 0 \text{ and } t_X(\langle e',u,a,z\rangle) \simeq 0] \\ &\simeq 1 \quad \text{if } \forall u[t_X(\langle e,a,u\rangle) \simeq 1 \text{ or } t_X(\langle e',u,a,z\rangle) \simeq 1] \end{aligned}$$

Application of F_i: $x = \langle\langle 16,n,e,i\rangle,a,z\rangle$.

$$\begin{aligned} t_X(x) &\simeq 0 \quad \text{if } \forall x'\exists y\ t_X(\langle e,x',a,y\rangle) \simeq 0 \text{ and} \\ &\qquad\qquad F_i(\lambda x'\{e\}^{\mathcal{L}}(x',a)) \simeq z \\ &\simeq 1 \quad \text{if } \exists x'\forall y\ t_X(\langle e,x',a,y\rangle) \simeq 1 \text{ or} \\ &\qquad\qquad [\forall x'\exists y\, t_X(\langle e,x',a,y\rangle) \simeq 0 \text{ and} \\ &\qquad\qquad \exists z'(F_i(\lambda x'\{e\}^{\mathcal{L}}(x',a)) \simeq z' \text{ and } z' = z)] \end{aligned}$$

If x does not look like a computation then $t_X(x) \simeq 1$. An index for t_X can be found by the second recursion theorem, hence t_X is partial recursive in $\mathcal{L}, X$, uniformly in X. By induction on the height of x in the tree the following can be proved: $x \in X \Rightarrow t_X(x)$ is defined, and $t_X(x) \simeq 0$ if x is a convergent computation and the part of the tree which lies below x is identical to the computation tree of x. $t_X(x) \simeq 1$ otherwise. Hence $R(X,x)$ iff $t_X(x) \simeq 0$ for all $x \in X$. Hence R is recursive in $\mathcal{L}$.

Suppose $x \in B$. Then $\{e_0\}^{\mathcal{L}}(x) \simeq 0$. Let X be the set of computations in the computation tree of $\{e_0\}^{\mathcal{L}}(x)$ ($\langle e_0,x,0\rangle$ included). Then X is recursive in $x, \mathcal{L}$ by lemma 10, and $R(X,x)$.

Hence $\exists X$ (X is recursive in $x, \mathcal{L}$ and $R(X,x)$).
If $\exists X\, R(X,x)$ then choose X such that $R(X,x)$. X is a set of convergent computations, and $\langle e_0, x, 0\rangle \in X$. Hence $\{e_0\}^{\mathcal{L}}(x) \simeq 0$, hence $x \in B$. This proves that

$$x \in B \iff \exists X\ (X \text{ is recursive in } x, \mathcal{L} \text{ and } R(X,x))$$
$$\iff \exists X\ R(X,x).$$

To prove the other direction of the equivalence in the theorem suppose $x \in B \iff \exists X$ (X is recursive in $x, \mathcal{L}$ and $R(X,x)$) where R is recursive in $\mathcal{L}$. Hence $x \in B \iff \exists m$ ($\lambda y\{m\}^{\mathcal{L}}(x,y)$ is a characteristic function and $R(X,x)$), where X is the set with characteristic function $\lambda y\{m\}^{\mathcal{L}}(x,y)$. Since $\mathcal{L}$ is normal the relations which are recursively enumerable in $\mathcal{L}$ are closed under the quantifier $\exists m$. Hence B is recursively enumerable in $\mathcal{L}$. $\square$

<u>Remark 1</u>: In this proof there are expressions of the form $\forall x\,(\ldots x, X, \ldots)$ where the expression inside the brackets is recursive in $\mathcal{L}$. The quantifier is expressed by E. This is permitted because E is recursive in $\mathcal{L}$. Weak recursiveness would not suffice.

Theorem 5 can be slightly strengthened. Let $\mathfrak{X}$ be a cartesian product where each factor is one of the following sets: A, the set of functions from A^n into S, the set of functionals. $\mathfrak{A} \subseteq \mathfrak{X}$ is recursively enumerable in $\mathcal{L}$ if there is an index m such that for all $\pi \in \mathfrak{X}$: $\pi \in \mathfrak{A} \iff \{m\}^{\mathcal{L},\pi}(0)\downarrow$.

<u>Theorem 6</u>: Suppose $\mathcal{L}$ is normal and that E is recursive in $\mathcal{L}$. Let $\mathfrak{A} \subseteq \mathfrak{X}$. Then $\mathfrak{A}$ is recursively enumerable in $\mathcal{L}$ iff there is a relation $R \subseteq P(A) \times \mathfrak{X}$ which is recursive in $\mathcal{L}$ such that for

all $\pi \in \mathfrak{X}$: $\pi \in \mathfrak{A} \Longleftrightarrow \exists X$ (X is recursive in $\mathcal{L},\pi$ and $(X,\pi) \in R$). The proof of theorem 6 is a slight modification of the proof of theorem 5.

By theorem 3 there are selection operators for numbers when $\mathcal{L}$ is normal. The next theorem states a similar result for S.

Theorem 7: Suppose $\mathcal{L}$ is normal. There is a function φ which is partial recursive in $\mathcal{L}$ with index $\hat{e}$ such that if $\exists x \in S\ \{e\}^{\mathcal{L}}(x,a)\downarrow$ then $\varphi(\langle e,a\rangle)\downarrow$ and $|\{\hat{e}\}^{\mathcal{L}}(\langle e,a\rangle)| \geq \min\{|\{e\}^{\mathcal{L}}(x,a)| : x \in S\}$. If $\varphi(\langle e,a\rangle)\downarrow$ then $\exists x \in S\ \{e\}^{\mathcal{L}}(x,a)\downarrow$.

Corollary: The relations which are recursively enumerable in $\mathcal{L}$ are closed under existential quantifiers over S.

Proof of theorem 7: The set $\{\langle e,x,a\rangle^{**} : x \in S\}$ is a family of elements in ${}^{S}\omega$ indexed by S. Hence the set can be coded by one element in ${}^{S}\omega$. Call this element α. Then

$$\exists s \in S\ \{e\}^{\mathcal{L}}(s,a)\downarrow \iff \exists s \in S\ \alpha[s]^{--} \in C^{\mathcal{L}},$$

where $C^{\mathcal{L}}$ is the set of convergent computations, and $\alpha[s] = \lambda r\, \alpha(\langle s,r\rangle)$.

Definition: For $\beta \in {}^{S}\omega$ let $\|\beta\| = \min\{|\beta[s]^{--}|^{\mathcal{L}} : s \in S\}$.

Lemma A: There is an index m such that

i) $\|\beta\| < \kappa^{\mathcal{L}} \implies \{m\}^{\mathcal{L}}(\beta)\downarrow$ and $\|\beta\| \leq |\{m\}^{\mathcal{L}}(\beta)|$,

ii) $\{m\}^{\mathcal{L}}(\beta)\downarrow \implies \|\beta\| < \kappa^{\mathcal{L}}$.

To prove theorem 7 it is enough to prove lemma A. The index $\hat{e}$ can easily be found from m.

Proof of lemma A : To find m we use the recursion theorem.
i) is proved by induction on $\|\beta\|$. Assume as induction hypothesis: $\|\beta\| < \mu \implies \{m\}^{\mathcal{L}}(\beta)\downarrow$ and $\|\beta\| < |\{m\}^{\mathcal{L}}(\beta)|$ for some ordinal μ. Assume $\|\alpha\| = \mu$.

Lemma 9 states that there is a relation $S(x,y)$ which is recursively enumerable in $\mathcal{L}$ such that if $x \in C^{\mathcal{L}}$ then $S(x,y)$ iff y is an immediate subcomputation of x. Let the relation R be defined by:

$$R(x,y,w) \iff S(x,y) \quad \text{if } x \text{ is not of the form } \langle\langle 10,n,e,e'\rangle,a\rangle\,,$$

$$R(x,y,w) \iff y = \langle e,a\rangle \text{ or } (w \in C^{\mathcal{L}} \text{ and } |\{e\}^{\mathcal{L}}(a)| \le |w| \text{ and } y = \langle e',\{e\}^{\mathcal{L}}(a),a\rangle) \quad \text{otherwise.}$$

Let $w \in C^{\mathcal{L}}$. If x is not a substitution then $\{y : R(x,y,w)\}$ is the set of immediate subcomputations of x. If x is a substitution then

$$\{y : R(x,y,w)\} = \begin{cases} \{\langle e,a\rangle\} & \text{if } |w| < |\{e\}^{\mathcal{L}}(a)| \\ \{\langle e,a\rangle,\langle e',\{e\}^{\mathcal{L}}(a),a\rangle\} & \text{otherwise.} \end{cases}$$

For $\sigma < \kappa^{\mathcal{L}}$ let T_σ be the relation defined by:

$$T_\sigma = \{\beta : \forall x \in S \;\; R(\alpha[x]^{--},\beta[x]^{--},w)\}\,,$$

where $|w| = \sigma$. Obviously

$$\beta \in T_\sigma \implies \|\beta\| < \|\alpha\|\,,$$

$$\sigma < \tau \implies T_\sigma \subseteq T_\tau\,.$$

T_σ is recursive in $\mathcal{L}, \alpha, w$ where $|w| = \sigma$, since R is recursive in $\mathcal{L}, w$ as a relation of x and y when $w \in C^{\mathcal{L}}$.

<u>Lemma B</u>: Let λ be an ordinal such that S is not cofinal in λ (i.e. there is no function $f : S \to \lambda$ such that $\lambda = \sup\{f(x) : x \in S\}$).

Let $\{\sigma(\tau) : \tau < \lambda\}$ be an increasing sequence of ordinals bounded above by $\kappa^{\mathcal{L}}$. Then there is an ordinal $\tau' < \lambda$ such that for all τ: $\tau' \leq \tau < \lambda \implies T_{\sigma(\tau)} = T_{\sigma(\tau')}$.

Proof: To obtain a contradiction suppose
$\forall \tau' < \lambda \; \exists \tau \; (\tau' \leq \tau < \lambda$ and $T_{\sigma(\tau')} \subsetneq T_{\sigma(\tau)})$.
Take $\tau' < \lambda$. Let τ be minimal such that $\tau' \leq \tau$ and $T_{\sigma(\tau')} \subsetneq T_{\sigma(\tau)}$. Obviously $\tau' < \tau$. Let $w', w \in C^{\mathcal{L}}$ be chosen such that $|w'| = \sigma(\tau)$. If $\beta \in T_{\sigma(\tau)} - T_{\sigma(\tau')}$ then $\forall x \in S \;\; R(\alpha[x]^{--}, \beta[x]^{--}, w)$ and $\neg \forall x \in S \;\; R(\alpha[x]^{--}, \beta[x]^{--}, w')$. Hence $\exists x \in S \; \neg R(\alpha[x]^{--}, \beta[x]^{--}, w')$. If $\neg R(\alpha[x]^{--}, \beta[x]^{--}, w')$ then $\alpha[x]^{--}$ is a substitution $\langle\langle 10, n, e, e'\rangle, a\rangle$, $\beta[x]^{--} = \langle s, \{e\}^{\mathcal{L}}(a), a\rangle$ and $|w'| < |\{e\}^{\mathcal{L}}(a)| \leq |w|$ (because $R(\alpha[x]^{--}, \beta[x]^{--}, w)$). Hence $R(\alpha[x]^{--}, \beta[x]^{--}, w'')$ for all $|w''| \geq |w|$. Let
$P(\tau') = \{x \in S : \exists \beta \in T_{\sigma(\tau)} - T_{\sigma(\tau')} \neg R(\alpha[x]^{--}, \beta[x]^{--}, w')\}$.

Then $P(\tau')$ is not empty, and $P(\tau') = P(\nu)$ for $\tau' \leq \nu < \tau$ because τ is minimal such that $T_{\sigma(\tau')} \subsetneq T_{\sigma(\tau)}$. If $\tau \leq \nu$ then $P(\tau')$ and $P(\nu)$ are disjoint, for if $x \in P(\tau')$ then $\alpha[x]^{--} = \langle\langle 10, n, e, e'\rangle, a\rangle$, $\beta[x]^{--} = \langle e', \{e\}^{\mathcal{L}}(a', a\rangle$ and $R(\alpha[x]^{--}, \beta[x]^{--}, w'')$ for all $w'' \in C^{\mathcal{L}}$ such that $\sigma(\tau) \leq |w''|$. Choose w'' such that $|w''| = \sigma(\nu)$. Then $R(\alpha[x]^{--}, \beta[x]^{--}, w'')$ since $\sigma(\tau) \leq \sigma(\nu)$. Hence $x \notin P(\nu)$. Let $f : S \to \lambda$ be defined by:

$$f(x) = \begin{cases} \text{the least } \tau' \text{ such that } x \in P(\tau') \\ \text{if } x \in \bigcup_{\tau<\lambda} P(\tau) \\ 0 \quad \text{otherwise} \end{cases}$$

Then $\sup\{f(x) : x \in S\} = \lambda$, a contradiction.

$\square$

Let $W \subseteq {}^S 2$ be the set of prewellorderings with domain $\subseteq S$. W is recursive in $\mathcal{L}$. For $\delta \in W$ let $Or(\delta)$ be the length of the prewellordering δ. Let $\lambda = \sup\{Or(\delta) : \delta \in W\}$. Then S is not cofinal in λ.

There is an index m_1 such that $\{m_1\}^{\mathcal{L}}(m,\alpha,w)\downarrow$ if $w \in C^{\mathcal{L}}$ and $\{m\}^{\mathcal{L}}(\beta)\downarrow$ for all $\beta \in T_{|w|}$, and if $w \in C^{\mathcal{L}}$ then $|\{m_1\}^{\mathcal{L}}(m,\alpha,w)| > |\{m\}^{\mathcal{L}}(\beta)|$ for all $\beta \in T_{|w|}$. By induction hypothesis $|\{m_1\}^{\mathcal{L}}(m,\alpha,w)| > \|\beta\|$ for all $\beta \in T_{|w|}$.

By the recursion theorem one can find an index m_2 such that $\{m_2\}^{\mathcal{L}}(m,\alpha,\gamma)\downarrow$ if $\gamma \in W$ and for all $\gamma' \in W$: $Or(\gamma') < Or(\gamma) \implies \{m_2\}^{\mathcal{L}}(m,\alpha,\gamma')\downarrow$ and $\{m_1\}^{\mathcal{L}}(m,\alpha,\langle m_2,m,\alpha,\gamma'\rangle)\downarrow$. When m_2 is chosen in the natural way the following is true:
$|\{m_2\}^{\mathcal{L}}(m,\alpha,\gamma)| > |\{m_2\}^{\mathcal{L}}(m,\alpha,\gamma')|, |\{m_1\}^{\mathcal{L}}(m,\alpha,\langle m_2,m,\alpha,\gamma'\rangle)|$ for all γ' such that $Or(\gamma') < Or(\gamma)$. Hence by induction hypothesis $|\{m_2\}^{\mathcal{L}}(m,\alpha,\gamma)| \geq \|\beta\|$ for all $\beta \in T_{|\langle m_2,m,\alpha,\gamma'\rangle|}$ when $Or(\gamma') < Or(\gamma)$.

There is an index m_3 such that $\{m_3\}^{\mathcal{L}}(m,\alpha)\downarrow$ if $\forall\gamma \in W\ \{m_2\}^{\mathcal{L}}(m,\alpha,\gamma)\downarrow$, and $|\{m_3\}^{\mathcal{L}}(m,\alpha)| > |\{m_2\}^{\mathcal{L}}(m,\alpha,\gamma)|$ for $\gamma \in W$.

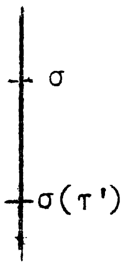
For $\tau < \lambda$ let
$\sigma(\tau) = \inf\{|\{m_2\}^{\mathcal{L}}(m,\alpha,\gamma)| : \gamma \in W \text{ and } Or(\gamma) = \tau\}$. Then $\{\sigma(\tau) : \tau < \lambda\}$ is a strictly increasing sequence of ordinals bounded above by $|\{m_3\}^{\mathcal{L}}(m,\alpha)|$, and hence by $\kappa^{\mathcal{L}}$. By lemma B there is an ordinal $\tau' < \lambda$ such that $T_{\sigma(\tau')} = T_{\sigma(\tau)}$ when $\tau' \leq \tau < \lambda$. Let $\sigma = \sup\{\sigma(\tau) : \tau < \lambda\}$.

<u>Claim</u>: $\sigma \geq \|\alpha\|$. (Hence $|\{m_3\}^{\mathcal{L}}(m,\alpha)| \geq \|\alpha\|$.)

To obtain a contradiction suppose $\sigma < \|\alpha\|$.

Let $x \in S$. If $\alpha[\overline{\overline{x]}}$ is a substitution $\langle\langle 10,n,e,e'\rangle,a\rangle$ then either $|\{e\}^{\mathcal{L}}(a)| < \sigma(\tau')$ or $\sigma \leq |\{e\}^{\mathcal{L}}(a)|$. For if

$\sigma(\tau') \leq |\{e\}^{\mathcal{L}}(a)| < \sigma$ take a $\beta \in T_{\sigma(\tau')}$. Let $\beta'[y] = \beta[y]$ if $y \neq x$, $\beta'[x] = \langle e', \{e\}^{\mathcal{L}}(a), a\rangle$. Then $\beta' \notin T_{\sigma(\tau')}$. But $\beta' \in T_{\sigma(\tau)}$ for $\sigma(\tau) > |\{e\}^{\mathcal{L}}(a)|$. This is contrary to the fact that $T_{\sigma(\tau)} = T_{\sigma(\tau')}$.



Let β be defined in the following way: If $\alpha[x]^{--}$ is not a substitution, let $\beta[x]^{--}$ be such that $S(\alpha[x]^{--})$ and $|\beta[x]^{--}| \geq \sigma$ (possible by assumption).

If $\alpha[x]^{--}$ is a substitution $\langle\langle 10, n, e, e'\rangle, a\rangle$ let

$$\beta[x]^{--} = \begin{cases} \langle s, \{e\}^{\mathcal{L}}(a), a\rangle & \text{if } |\{e\}^{\mathcal{L}}(a)| < \sigma(\tau') \\ \langle e, a\rangle & \text{if } |\{e\}^{\mathcal{L}}(a)| \geq \sigma \end{cases}$$

Then $\|\beta\| \geq \sigma$. $\|\beta\| = \inf\{|\beta[x]^{--}| : x \in S\}$. Obviously $|\beta[x]^{--}| \geq \sigma$ in all cases except for $|\{e\}^{\mathcal{L}}(a)| < \sigma(\tau')$. In this case $|\{e'\}^{\mathcal{L}}(\{e\}^{\mathcal{L}}(a), a)| \geq \sigma$ because otherwise $|\alpha[x]^{--}| \leq \sigma$, contrary to the assumption. Hence $|\beta[x]^{--}| \geq \sigma$ also in this case. Hence $\beta \notin T_{\sigma(\tau')}$. For suppose $\beta \in T_{\sigma(\tau')}$. Choose τ such that $\tau' < \tau < \lambda$. Choose $\gamma', \gamma \in W$ such that $Or(\gamma') = \tau'$, $Or(\gamma) = \tau$, $\sigma(\tau') = |\{m_2\}^{\mathcal{L}}(m, \alpha, \gamma')|$, $\sigma(\tau) = |\{m_2\}^{\mathcal{L}}(m, \alpha, \gamma)|$. By the construction of m_2 $|\{m_2\}^{\mathcal{L}}(m, \alpha, \gamma)| \geq \|\beta'\|$ for all $\beta' \in T_{|\langle m_2, m, \alpha, \gamma'\rangle|}$. Hence $|\{m_2\}^{\mathcal{L}}(m, \alpha, \gamma)| \geq \|\beta\|$ since $\beta \in T_{\sigma(\tau')} = T_{|\langle m_2, m, \alpha, \gamma'\rangle|}$, contradicting the fact that $|\{m_2\}^{\mathcal{L}}(m, \alpha, \gamma)| = \sigma(\tau) < \sigma \leq \|\beta\|$.

By the construction of β $\beta \in T_{\sigma(\tau')}$, a contradiction. This proves the claim.

By the second recursion theorem there is an m such that

$$\{m_3\}^{\mathcal{L}}(m, \alpha) \simeq \{m\}^{\mathcal{L}}(\alpha),$$
$$|\{m\}^{\mathcal{L}}(\alpha)| \geq |\{m_3\}^{\mathcal{L}}(m, \alpha)|$$

for all α. This m satisfies part i) of lemma A.
By the induction hypothesis mentioned in the beginning of the proof:

$$\|\beta\| < \mu \implies \{m\}^{\mathcal{L}}(\beta)\downarrow \text{ and } \|\beta\| \leq |\{m\}^{\mathcal{L}}(\beta)| .$$

Let $\|\alpha\| = \mu$. By the claim $\|\alpha\| \leq |\{m_3\}^{\mathcal{L}}(m,\alpha)|$. Hence $\|\alpha\| \leq |\{m\}^{\mathcal{L}}(\alpha)|$.

Part ii) of lemma A can be proved by induction on the length of $|\{m\}^{\mathcal{L}}(\beta)|$. The induction goes as follows.

Suppose $\{m\}^{\mathcal{L}}(\alpha)\downarrow$ and ii) is satisfied for all β such that $|\{m\}^{\mathcal{L}}(\beta)| < |\{m\}^{\mathcal{L}}(\alpha)|$. Since $\{m\}^{\mathcal{L}}(\alpha)\downarrow$, $\{m_3\}^{\mathcal{L}}(m,\alpha)\downarrow$ and $|\{m\}^{\mathcal{L}}(\alpha)| > |\{m_3\}^{\mathcal{L}}(m,\alpha)|$. Also $\{m_2\}^{\mathcal{L}}(m,\alpha,\gamma)\downarrow$ for all $\gamma \in W$, and $|\{m_3\}^{\mathcal{L}}(m,\alpha)| > |\{m_2\}^{\mathcal{L}}(m,\alpha,\gamma)|$ for all $\gamma \in W$.
Let the ordinals $\{\sigma(\tau) : \tau < \lambda\}$ be defined as before. Choose $\tau' < \lambda$ as before. If $\alpha[x]^{--}$ is a substitution, then either $|\{e\}^{\mathcal{L}}(a)| < \sigma(\tau')$ or $\sigma \leq |\{e\}^{\mathcal{L}}(a)|$ by the argument in the proof of the claim. To obtain a contradiction suppose $\|\alpha\| = \varkappa^{\mathcal{L}}$ (i.e. $\alpha[x]^{--}$ codes a divergent computation for all $x \in S$). Construct β as follows: If $\alpha[x]^{--}$ is not a substitution let $\beta[x]^{--}$ be a divergent subcomputation of $\alpha[x]^{--}$. If $\alpha[x]^{--}$ is a substitution let $\beta[x]^{--}$ be defined as in the proof of the claim. Then by construction $\beta \in T_{\sigma(\tau')}$, and it is easy to check that $|\beta[x]^{--}| \geq \sigma$ for all $x \in S$. Hence $\|\beta\| \geq \sigma$. Choose $\gamma' \in W$ such that $\sigma(\tau') = |\{m_2\}^{\mathcal{L}}(m,\alpha,\gamma')|$. Choose $\gamma \in W$ such that $Or(\gamma') < Or(\gamma)$. Then by the construction of m_2:

$$|\{m_2\}^{\mathcal{L}}(m,\alpha,\gamma)| > |\{m_1\}^{\mathcal{L}}(m,\alpha,\langle m_2,m,\alpha,\gamma'\rangle)| .$$

By the construction of m_1: $|\{m_1\}^{\mathcal{L}}(m,\alpha,\langle m_2,m,\alpha,\gamma'\rangle)| > |\{m\}^{\mathcal{L}}(\beta)|$ since $\beta \in T_{|\langle m_2,m,\alpha,\gamma'\rangle|} = T_{\sigma(\tau')}$. Hence $|\{m\}^{\mathcal{L}}(\alpha)| > |\{m\}^{\mathcal{L}}(\beta)|$.
By the induction hypothesis $\|\beta\| < \varkappa^{\mathcal{L}}$. By part i) of lemma A: $\|\beta\| \leq |\{m\}^{\mathcal{L}}(\beta)|$. Hence $\|\beta\| < |\{m_2\}^{\mathcal{L}}(m,\alpha,\gamma)|$ for all $\gamma \in W$ such

that $Or(\gamma') < Or(\gamma)$. By the definition of $\sigma(\tau)$ $\|\beta\| < \sigma(\tau)$ when $\tau' < \tau < \lambda$. Hence $\|\beta\| < \sigma$, contradicting the fact that $\|\beta\| \geq \sigma$. This proves lemma A. $\square$

§ 5 KLEENE RECURSION IN NORMAL OBJECTS OF TYPE $n+2$, $n>0$

Let F be an object of type $n+2$. Let $S = Tp(0) \cup Tp(1) \cup \ldots \ldots \cup Tp(n-1)$. Let $\mathcal{S}$ be the coding scheme from § 3. It is proved in § 3 that there is a list $\mathcal{L}$ such that recursion in $\mathcal{L}$ is essentially the same as Kleene recursion in F. There are primitive recursive functions f, g such that

$$\begin{aligned} \{e\}^F(a) &\simeq m \iff \{f(e)\}^{\mathcal{L}}(a) \simeq m, \\ |\{e\}^F(a)|^F &\leq |\{f(e)\}^{\mathcal{L}}(a)|^{\mathcal{L}}, \\ \{e\}^{\mathcal{L}}(a) &\simeq m \iff \{g(e)\}^F(a) \simeq m, \\ |\{e\}^{\mathcal{L}}(a)|^{\mathcal{L}} &\leq |\{g(e)\}^F(a)|^F \end{aligned}$$

for all e,a,m, where $e,m \in \omega$, a is a list of elements in $\bigcup_{i \leq n} Tp(i)$. (A part of this is stated in lemma 4 and lemma 5.)

As mentioned in § 4 $\mathcal{L}$ is normal iff ${}^{n+2}E$ is weakly recursive in F. In this chapter we will deduce results about Kleene recursion in F from the results in § 4. Let us start with theorem 2.

Let $C^F = \{\langle e,a\rangle : \{e\}^F(a)\downarrow\}$.

Then $\langle e,a\rangle \in C^F \iff \{e\}^F(a)\downarrow \iff \{f(e)\}^{\mathcal{L}}(a)\downarrow$. There is a function $Ord : Tp(n) \to$ Ordinals defined by:

$$Ord(\langle e,a\rangle) = |\{f(e)\}^{\mathcal{L}}(a)|^{\mathcal{L}}$$

Let $x = \langle e,a\rangle$, $y = \langle e',a'\rangle$. Let $p'(x,y) \simeq p(\langle f(e),a\rangle,\langle f(e'),a'\rangle)$ where p is the function from theorem 2. p' is partial recursive in F in the sense of Kleene by lemma 5. This proves

<u>Theorem 2'</u>: Suppose ${}^{n+2}E$ is weakly recursive in F. Then there is a function p' which is partial recursive in F such that

$$x \in C^F \text{ or } y \in C^F \iff p'(x,y)\downarrow ,$$

$$x \in C^F \text{ and } \mathrm{Ord}(x) \leq \mathrm{Ord}(y) \implies p'(x,y) \simeq 0 ,$$

$$\mathrm{Ord}(x) > \mathrm{Ord}(y) \implies p'(x,y) \simeq 1 .$$

<u>Theorem 3'</u>: Suppose ${}^{n+2}E$ is weakly recursive in F. Then there is a function φ' which is partial recursive in F such that for all e, a: If $\exists n \in N\ \{e\}^F(n,a)\downarrow$ then $\varphi'(e,a)\downarrow$, and $\{e\}^F(\varphi'(e,a),a)\downarrow$. If $\varphi'(e,a) \simeq n$ then $\{e\}^F(n,a)\downarrow$.

Proof: Let $\varphi'(e,a) = \varphi(f(e),a)$, where φ is the selection operator from theorem 3. φ' is partial recursive in F by lemma 5. $\square$

<u>Theorem 4'</u>: Suppose ${}^{n+1}E$ is weakly recursive in F. Then there is a relation R' which is recursively enumerable in F such that for all e, a:

$$\{e\}^F(a)\uparrow \iff \exists \alpha \in Tp(n)\ R'(\alpha,\langle e,a\rangle) .$$

Proof: The equality relation on S is recursive in $\mathcal{L}$, and the functional E_S is weakly recursive in $\mathcal{L}$ since ${}^{n+1}E$ is weakly recursive in F. By theorem 4 there is a relation R which is recursively enumerable in $\mathcal{L}$ such that for all n, b: $\{n\}^{\mathcal{L}}(b)\uparrow \iff \exists x\, R(x,\langle n,b\rangle)$. Hence $\{e\}^F(a)\uparrow \iff \{f(e)\}^{\mathcal{L}}(a)\uparrow \iff \exists x\, R(x,\langle f(e),a\rangle)$. Let p be the function: $A \to Tp(n)$, and $q: A \to A$ the inverse of p, defined in § 3. Then $\exists x\, R(x,\langle f(e),a\rangle) \iff \exists \alpha \in Tp(n)\ R(q(\alpha),\langle f(e),a\rangle$. Let $R'(\alpha,\langle e,a\rangle) \iff R(q(\alpha),\langle f(e),a\rangle)$. $\square$

Remark: Theorem 4' is a slightly weaker result than corollary 5.2 in [13], where the assumption " ${}^{n+1}E$ is weakly recursive in F " is omitted. Here theorem 4' is a corollary of theorem 4, which is proved for arbitrary sets S. To describe the functions which code elements in A as elements in ${}^{S}\omega$ we need the equality relation on S and the functional E_S. When working in the type hiearchy the structure is so rich that it is not necessary to introduce E_S. Hence a better result can be obtained in that case.

Theorem 5': Suppose E is recursive in F. Let $B \subseteq Tp(n)$. B is recursively enumerable in F iff there is a relation $R' \subseteq P(Tp(n)) \times Tp(n)$ which is recursive in F such that for all $x \in Tp(n)$: $x \in B \iff \exists X$ (X is recursive in x, F and $R'(X,x)$).

Proof: When $\mathcal{L}$ is normal Tp(n) is a recursive subset of A. Let $B \subseteq Tp(n)$. B is recursively enumerable in F iff B is recursively enumerable in $\mathcal{L}$. Suppose B is recursively enumerable in F. By theorem 5 there is a relation $R \subseteq P(A) \times A$ which is recursively enumerable in $\mathcal{L}$ such that for all $x \in A$: $x \in B$ $\iff \exists X$ (X is recursive in $x, \mathcal{L}$ and $R(X,x)$). Let R' be defined by: $R'(Y,y) \iff R(q[Y],y)$, where $Y \subseteq Tp(n), y \in Tp(n)$. Then R' is recursive in $\mathcal{L}$, hence in F. If q[Y] is recursive in $\mathcal{L}, x$ then Y is, and Y is recursive in F, x. Hence $x \in B$ $\iff \exists Y$ (Y is recursive in F, x and $R'(Y,y)$). The other part of theorem 5' can be proved as follows: Suppose $x \in B \iff$ $\exists X$ (X is recursive in x, F and $R'(X,x)$) where R' is recursive in F. Then R' is recursive in $\mathcal{L}$, and X is recursive in x, F iff X is recursive in $\mathcal{L}$. Hence $x \in B \iff \exists X$ (X is recursive in $x, \mathcal{L}$ and $R'(X,x)$). By theorem 5 B is recursively enumerable in $\mathcal{L}$, and hence in F.

□

<u>Theorem 6'</u>: Suppose ${}^{n+2}E$ is recursive in F. Let $\mathfrak{A} \subseteq Tp(n+2)$. Then $\mathfrak{A}$ is recursively enumerable in F iff there is a relation $R' \subseteq P(Tp(n)) \times Tp(n+2)$ which is recursive in F such that for all $G \in Tp(n+2)$: $G \in \mathfrak{A} \iff \exists X\,(X$ is recursive in F, G and $R(X,G))$.

<u>Theorem 7'</u>: Suppose ${}^{n+2}E$ is weakly recursive in F. Then there is a function φ' which is partial recursive in F with index $\hat{e}'$ such that if $\exists x \in Tp(n-1)\ \{e\}^F(x,a)\downarrow$ then $\varphi'(\langle e,a\rangle)\downarrow$ and $|\{\hat{e}'\}^F(\langle e,a\rangle)| \geq \min\{|\{e\}^F(x,a)| : x \in Tp(n-1)\}$. Moreover if $\varphi'(\langle e,a\rangle)\downarrow$ then $\exists x \in Tp(n-1)\ \{e\}^F(x,a)\downarrow$.

Proof: Let φ be the function from theorem 7 with index $\hat{e}$. If $\exists x \in Tp(n-1)\ \{e\}^F(x,a)\downarrow$ then $\exists x \in Tp(n-1)\ \{f(e)\}^{\mathcal{L}}(x,a)\downarrow$, hence $\exists x \in S\ \{f(e)\}\ (x,a)\downarrow$. Then $\varphi(\langle f(e),a\rangle)\downarrow$, and $|\{\hat{e}\}^{\mathcal{L}}(\langle f(e),a\rangle)|$
$\geq \inf\{|\{f(e)\}^{\mathcal{L}}(x,a)| : x \in S\}$
$\geq \inf\{|\{e\}^F(x,a)| : x \in Tp(n-1)\}$. Let $s = g(\hat{e})$. Then $\{s\}^F(\langle f(e),a\rangle) \simeq \{\hat{e}\}^{\mathcal{L}}(\langle f(e),a\rangle)$, and $|\{s\}^F(\langle f(e),a\rangle)|$
$\geq |\{\hat{e}\}^{\mathcal{L}}(\langle f(e),a\rangle)|$. Choose $\hat{e}'$ such that $\{\hat{e}'\}^F(\langle e,a\rangle)$
$\simeq \{s\}^F(\langle f(e),a\rangle)$ and $|\{\hat{e}'\}^F(\langle e,a\rangle)| \geq |\{s\}^F(\langle f(e),a\rangle)|$. Let $\varphi'(\langle e,a\rangle) \simeq \{\hat{e}'\}^F(\langle e,a\rangle)$.

§ 6 COMPUTATION THEORIES ON $\mathfrak{A}$

A computation theory on $\mathfrak{A}$ is a pair $(\Theta, ||_\Theta)$ such that the following is true:

Θ is a set of tuples (e,a,r) where $e \in N$, a is a list of objects from A, $r \in S$. $||_\Theta$ is a function from Θ onto some ordinal $\varkappa_\Theta$. If $(e,a,r) \in \Theta$ and $(e,a,r') \in \Theta$ then $r = r'$. Let $\{e\}_\Theta$ denote the partial function defined by

$$\{e\}_\Theta(a) \simeq r \iff (e,a,r) \in \Theta$$

φ is Θ-computable if there is an e such that $\varphi = \{e\}_\Theta$, in which case e is said to be an index for φ.

The following functions are Θ-computable: the characteristic functions of N and S, the constant functions $f(a) = n$ $(n \in N)$, M, K, L, i, s, f where

$$i(x,a) = \begin{cases} x & \text{if } x \in S \\ 0 & \text{otherwise} \end{cases}$$

$$s(x) = \begin{cases} x+1 & \text{if } x \in N \\ 0 & \text{otherwise} \end{cases}$$

$$f(x,y) = \begin{cases} x(y) & \text{if } x \in {}^{S}\omega,\ y \in S \\ 0 & \text{otherwise} \end{cases}$$

The following operations are allowed: substitution, primitive recursion, permutations of the list of arguments of a function, adding dummy arguments, substitution of a function for an element in ${}^{S}\omega$, diagonalization, the S^n_m-property is satisfied. To make precise what is meant by "an operation is allowed" let us regard substitution, diagonalization and the S^n_m-property.

Substitution: There is a Θ-computable mapping $g_1(e,f,n)$ such that for all e,f,a,x:

$\{g_1(e,f,n)\}_\Theta(a) \simeq x \iff \exists u[\{e\}_\Theta(a) \simeq u$ and $\{f\}_\Theta(u,a) \simeq x]$

and $|g_1(e,f,n),a,x|_\Theta \geq \sup\{|e,a,u|_\Theta + 1, |f,u,a,x|_\Theta + 1\}$

where $\{e\}_\Theta(a) \simeq u$ and $\{f\}_\Theta(u,a) \simeq x$. The list a has length n.

Diagonalization: There is a Θ-computable mapping $g_2(m,n)$ such that for all e,a,b,x:

$\{g_2(m,n)\}_\Theta(e,a,b) \simeq x \iff \{e\}_\Theta(a) \simeq x$, and

$|g_2(m,n),e,a,b,x|_\Theta \geq |e,a,x|_\Theta + 1$ when $\{e\}_\Theta(a) \simeq x$. The lists a and b have lengths m and n respectively.

S^n_m-property: There is a Θ-computable mapping $g_3(n,m)$ such that $g_3(n,m)$ is an index for a mapping S^n_m with the following property: For all $e \in N$, $x_1 \dots x_n \in N$, $y_1 \dots y_m, z$

$\{S^n_m(e,x_1\dots x_n)\}_\Theta(y_1\dots y_m) \simeq z \iff \{e\}_\Theta(x_1\dots x_n,y_1\dots y_m) \simeq z$,

and $|S^n_m(e,x_1\dots x_n),y_1\dots y_m,z|_\Theta \geq |e,x_1\dots x_n,y_1\dots y_m,z|_\Theta + 1$,

when $\{e\}_\Theta(x_1\dots x_n,y_1\dots y_m) \simeq z$.

(A mapping is a function which is totally defined.)

This ends the definition of a computation theory on $\mathfrak{A}$.

Let $\mathcal{L}$ be a list of relations, functions and functionals. Let $\Theta = \{(e,a,r) : \{e\}^{\mathcal{L}}(a) \simeq r\}$, and let $|e,a,r|_\Theta = |\{e\}^{\mathcal{L}}(a)|^{\mathcal{L}}$. Then $(\Theta, ||_\Theta)$ is a computation theory on $\mathfrak{A}$.

Some notations and definitions:

Let $(\Theta, ||_\Theta)$ be a computation theory on $\mathfrak{A}$. A <u>computation</u> is a tuple (e,a,r). The computation is <u>convergent</u> if $(e,a,r) \in \Theta$. Otherwise it is <u>divergent</u>. If $(e,a,r) \in \Theta$ then $|e,a,r|_\Theta$ is called the <u>length</u> of the computation (e,a,r). The expression

"$\{e\}_\Theta(a)\downarrow$" is an abbreviation for the statement: there is an r such that $\{e\}_\Theta(a) \simeq r$. "$\{e\}_\Theta(a)\uparrow$" is an abbreviation for the negation of this statement. If $\{e\}_\Theta(a)\downarrow$ then there is a unique r such that $\{e\}_\Theta(a) \simeq r$. Hence there is no ambiguity in denoting the computation by $\langle e,a\rangle$. Sometimes it will be denoted by $\{e\}_\Theta(a)$. Hence $\{e\}_\Theta(a)$ has a double meaning. It denotes the object r such that $\{e\}_\Theta(a) \simeq r$, and also the computation (e,a,r). Let $|\{e\}_\Theta(a)|_\Theta = |\langle e,a\rangle|_\Theta = |e,a,r|_\Theta$ where $\{e\}_\Theta(a) \simeq r$. If there is no r such that $\{e\}_\Theta(a) \simeq r$ let $|\{e\}_\Theta(a)|_\Theta = |\langle e,a\rangle|_\Theta = \varkappa_\Theta$, where $\varkappa_\Theta = \sup\{|\{e\}_\Theta(a)|_\Theta : \{e\}_\Theta(a)\downarrow\}$.

The operations (substitution, diagonalization, ...) have the following property: If we start with some computations and perform an operation then we obtain computations with greater length than the original ones. This corresponds to the intuitive picture of a computation, where the length is a measure of how many operations one must do to obtain a result. The more operations we do, the greater the length will be.

Let $\mathcal{F}$ be a partial monotone functional. $\mathcal{F}$ is <u>weakly Θ-computable</u> if there is a Θ-computable mapping $g(e,n)$ such that for all $e \in N$, $r \in S$, all lists a of length n: $\{g(e,n)\}_\Theta(a) \simeq r \iff \lambda x\{e\}_\Theta(x,a) \in \operatorname{dom}\mathcal{F}$ and $\mathcal{F}(\lambda x\{e\}_\Theta(x,a)) \simeq r$. Moreover if $\{g(e,n)\}_\Theta(a) \simeq r$ then there is a subfunction ψ of $\lambda x\{e\}_\Theta(x,a)$ such that $\mathcal{F}(\psi) \simeq r$ and $|\{g(e,n)\}_\Theta(a)|_\Theta > |\{e\}_\Theta(x,a)|_\Theta$ for all $x \in \operatorname{dom}\psi$. A <u>$\Theta$-index</u> for $\mathcal{F}$ is an index for the mapping g.

If F is a functional defined on total functions then by the above definition F is weakly Θ-computable if there is a Θ-computable mapping $g(e,n)$ such that for all $e \in N$, $r \in S$, all lists a of length n: $\{g(e,n)\}_\Theta(a) \simeq r \iff \lambda x\{e\}_\Theta(x,a)$ is total and

$F(\lambda x\{e\}_\Theta(x,a)) \simeq r$. If $\{g(e,n)\}_\Theta(a)\downarrow$ then $|\{g(e,n)\}_\Theta(a)|_\Theta >$ $|\{e\}_\Theta(x,a)|_\Theta$ for all $x \in A$.

First recursion theorem: Suppose $\mathcal{F}$ is a partial monotone functional, $\mathcal{F}$ is weakly Θ-computable, and the argument list of $\mathcal{F}$ has the form φ, a, where φ is ranging over k-ary partial functions, and a is ranging over A^k. Then there is a least solution φ to the equality $a[\mathcal{F}(\varphi,a) \simeq \varphi(a)]$, and this least solution is Θ-computable.

Second recursion theorem: $e\ x\ a\ \{e\}_\Theta(x,a) \simeq \{x\}_\Theta(a)$.

The next two definitions are inspired by Moschovakis: Axioms for Computation Theories - First Draft ([16]). In this paper a subset X of the domain is said to be finite in a computation theory Θ if the relations which are Θ-semicomputable are closed under the quantifiers $\forall x \in X$, $\exists x \in X$ in a uniform way. Below follow two different notions of finitiness. The first one is the same as the one defined by Moschovakis [16].

Let X be a subset of A. X is strongly Θ-finite if the partial functional $\mathcal{F}_X$ defined by

$$\mathcal{F}_X(\varphi) \simeq \begin{cases} 0 & \text{if } \exists x \in X \ \ \varphi(x) \simeq 0 \\ 1 & \text{if } \forall x \in X \ \ \exists r \neq 0 \ \ \varphi(x) \simeq r \end{cases}$$

is weakly Θ-computable. X is weakly Θ-finite if the functional F_X defined by

$$F_X(f) = \begin{cases} 0 & \text{if } \exists x \in X \ \ f(x) = 0 \\ 1 & \text{if } \forall x \in X \ \ \exists r \neq 0 \ \ f(x) = r \end{cases}$$

is weakly Θ-computable. (φ ranges over partial functions, f

over functions defined on all of X.) If e is an index for $\mathcal{F}_X$ (F_X) we say that e proves that X is strongly (weakly) finite.

<u>Remark</u>: i) A is weakly Θ-finite $\Longleftrightarrow$ E is weakly Θ-computable.

ii) If $X \subseteq A$ is weakly Θ-finite then the Θ-computable relations on A are closed under the quantifiers $\exists x \in X$, $\forall x \in X$.

iii) If $X \subseteq A$ is strongly Θ-finite then the Θ-semicomputable relations on A are closed under the quantifiers $\exists x \in X$, $\forall x \in X$.

<u>Lemma 11</u>: If $X \subseteq A$ is strongly Θ-finite then X is weakly Θ-finite.

Lemma 12: Suppose that the equality relation on A is Θ-computable. Let $*$ denote one of the following two properties: "strongly Θ-finite", "weakly Θ-finite". Suppose $X \subseteq A$ is $*$. Then i) - iv) are true.

i) X is Θ-computable,

ii) If $Y \subseteq X$ is Θ-computable then Y is $*$.

iii) If there is a Θ-computable mapping g such that for all $x \in X$ $g(x)$ is an index which proves that a set Y_x is $*$ then $\bigcup_{x \in X} Y_x$ and $\bigcap_{x \in X} Y_x$ are $*$.

iv) If f is a Θ-computable mapping then $f[X]$ (the image of X under f) is $*$.

<u>Lemma 13</u>: Let $X = \{x_1 \ldots x_n\}$ where $x_1 \ldots x_n \in A$. Then X is weakly Θ-finite in $x_1 \ldots x_n$. If Θ admits selection operators for natural numbers then X is also strongly Θ-finite in $x_1 \ldots x_n$.

Definitions: Θ admits selection operators for natural numbers if there is a Θ-computable mapping $g(e,n)$ such that for all $e,n \in N$, all lists a of length n: If $\exists m \in N \quad \{e\}_\Theta(m,a) \simeq 0$ then $\{g(e,n)\}_\Theta(a)\downarrow$ and has a value in N and $\{e\}_\Theta(\{g(e,n)\}_\Theta(a),a) \simeq 0$, and $|\{g(e,n)\}_\Theta(a)| > \inf\{|\{e\}_\Theta(m,a)| : \{e\}_\Theta(m,a) \simeq 0\}$. If $\{g(e,n)\}_\Theta(a) \simeq m$ then $\{e\}_\Theta(m,a) \simeq 0$.

A relation R on A is Θ-computable if the characteristic function of R is Θ-computable. R is Θ-computable in b if there is an index e such that $\lambda a\ \{e\}_\Theta(a,b)$ is the characteristic function of R. R is Θ-semicomputable if there is an index e such that for all a: $R(a) \iff \{e\}_\Theta(a)\downarrow$. R is Θ-semicomputable in b if $R(a) \iff \{e\}_\Theta(a,b)\downarrow$ for some $e \in N$. A partial function φ is Θ-computable in b if $\varphi = \lambda a\ \{e\}_\Theta(a,b)$ for some e. A partial monotone functional $\mathcal{F}$ is weakly Θ-computable in b if there is a Θ-computable mapping $g(e,n)$ such that $\mathcal{F}(\lambda x\{e\}_\Theta(x,a)) \simeq \{g(e,n)\}_\Theta(a,b)$. If $\{g(e,n)\}_\Theta(a,b)\downarrow$ then there is a subfunction ψ of $\lambda x\ \{e\}_\Theta(x,a)$ such that $\mathcal{F}(\psi) \simeq \{g(e,n)\}_\Theta(a,b)$ and $|\{g(e,n)\}_\Theta(a,b)| > |\{e\}_\Theta(x,a)|$ for all $x \in \mathrm{dom}\,\psi$. A set X is strongly (weakly) Θ-finite in b if $\mathcal{F}_X$ (F_X) is weakly Θ-computable in b.

Let $C_\Theta = \{\langle e,a\rangle : \{e\}_\Theta(a)\downarrow\}$. $|\,|_\Theta$ is a Θ-norm if there is a partial Θ-computable function $p(x,y)$ such that $p(x,y) \simeq 0$ if $x \in C_\Theta$ and $|x|_\Theta \le |y|_\Theta$, $p(x,y) \simeq 1$ if $|x|_\Theta > |y|_\Theta$. $(\Theta, |\,|_\Theta)$ is p-normal if $|\,|_\Theta$ is a Θ-norm.

Lemma 14: If $|\,|_\Theta$ is a Θ-norm then Θ admits selection operators for natural numbers.

Proof: Same proof as for theorem 3.

<u>Lemma 15</u>: Let $\mathscr{L}$ be a normal list, let Θ denote recursion in $\mathscr{L}$, and let $||_\Theta$ be the natural length function. Then A is weakly Θ-finite, S is strongly Θ-finite and $||_\Theta$ is a Θ-norm.

Proof by theorems 2 and 7.

§ 7 ABSTRACT KLEENE THEORIES

Let F be an object of thpe $n+2$. Let Θ be the set of tuples (e,a,n) such that $\{e\}^F(a) \simeq n$ in the sense of Kleene. Let $||_\Theta$ denote the natural length function. Then $\langle \Theta, ||_\Theta \rangle$ is not a computation theory in the sense defined in § 6 for some trivial reasons. See the discussion in § 3. The following question arises: Is there a computation theory which is "similar" to $\langle \Theta, ||_\Theta \rangle$? This is analoguous to the problems in § 3.

We introduce the notion of an abstract Kleene theory. $\langle \Theta, ||_\Theta \rangle$ defined above is an example of such a theory. When $S = Tp(0) \cup \dots$ $\dots \cup Tp(n-1)$ there is a close connection between Kleene recursion in a list of objects and recursion in a list $\mathcal{L}$ on $\mathfrak{A}$. The motivation for the new notion is to have structures which are related to computation theories on $\mathfrak{A}$ in the same way as Kleene recursion in a list of objects is related to recursion in $\mathcal{L}$.

An abstract Kleene theory on $(Tp(0),\dots,Tp(n))$ is a pair $\langle \Theta, ||_\Theta \rangle$ where Θ is a set of tuples (e,a,n) of length ≥ 2. $e,n \in \omega$, a is a list of objects from $\bigcup_{i \leq n} Tp(i)$. If $(e,a,n) \in \Theta$ and $(e,a,n') \in \Theta$ then $n = n'$. A partial function φ is Θ-computable with index e if for all a,n: $\varphi(a) \simeq n \iff (e,a,n) \in \Theta$. φ is also denoted by $\{e\}_\Theta$. If φ is Θ-computable then the domain of φ is a subset of a cartesian product $Tp(i_1) \times \dots \times Tp(i_k)$ where $0 \leq i_j \leq n$ for $1 \leq j \leq k$. The following functions are Θ-computable: $s(n,a) = n+1$, $f(a) = n$, $f(n,a) = n$, $f(n,\alpha,a) = \alpha(n)$ $(\alpha \in Tp(1))$. The following operations are allowed: substitution, primitive recursion, permutations of a list of arguments, diagonalization. The S^n_m-property is satisfied, and the functional F_i is weakly Θ-computable, where $F_i(\alpha,f) = \alpha(\lambda x \in Tp(i) f(x))$,

$\alpha \in Tp(i+1)$, $i < n$. The operations satisfy certain ordinal inequalities similar to those given in the definition of a computation theory.

The main difference between a computation theory and an abstract Kleene theory is that in the latter all computable functions have values in N, and the domain is a subset of a cartesian product of types. In the former theory the computable functions have values in S $(= Tp(0) \cup \ldots \cup Tp(n-1))$, and the domain is a subset of A^k for some k.

<u>Examples</u> of abstract Kleene theories: Let F be an object of type $> n+2$. Let $\Theta = \{(e,a,n) : \{e\}^F (a) \simeq n$, $e,n \in N$, a is a list of objects of type $\leq n\}$. In this case the "natural" Kleene theory is $\{(e,a,n) : \{e\}^F(a) \simeq n$, a is a list of objects of type $\leq k\}$, where the type of F is $k+2 > n+2$. For the latter theory there is a natural length function $||^F$. A length function $||_\Theta$ for Θ can be constructed from $||^F$. If F is a normal object then $||_\Theta$ is a Θ-norm, and $Tp(i)$ is strongly Θ-finite for $0 \leq i \leq n$.

Let S be the object of type $n+3$ defined by

$$S(e,F) = \begin{cases} 0 & \text{if } \{e\}^F(0)\downarrow \\ 1 & \text{if } \{e\}^F(0)\uparrow \end{cases}$$

where $e \in N$, $F \in Tp(n+2)$. In [7] Harrington constructs a hierarchy for the functions which are recursive in S, F. An abstract Kleene theory can be obtained from this hierarchy. This theory is p-normal, $Tp(n)$ is weakly (not strongly) finite, $Tp(n-1)$ is strongly finite.

Let $S = Tp(0) \cup \ldots \cup Tp(n-1)$. Let $\langle \Theta, ||_\Theta \rangle$ be an abstract Kleene theory on $\mathfrak{A}$. Then there is a computation theory $(\Psi, ||_\Psi)$

which is similar to $\langle \Theta, ||_\Theta \rangle$. $(\Psi, ||_\Psi)$ is defined as follows: Let $i = \langle i_1 \dots i_k \rangle$ where $0 \leq i_j \leq n$ for $j = 1 \dots k$, and let φ_i be the partial function defined by: $\varphi_i(e,a) \simeq \{e\}_\Theta(a)$, where e ranges over N, a over $Tp(i_1) \times \dots \times Tp(i_k)$. Then φ_i is Θ-computable. Let $\mathcal{L}$ be the list of the φ_i's and of the functions and functionals which expresses the type structure (i.e. $g_1, g_2, G, G_1, \dots, G_{n-1}$ in § 3). Let Ψ be the set of convergent computations generated from $\mathcal{L}$. Let $||_\Psi$ be the function defined by: $|e,a,r|_\Psi = 0$ if $(e,a,r) \in \Gamma(\emptyset)$ by some other clause than the clauses for the φ_i's. (Γ is the operator which generates the convergent computations in $\mathcal{L}$). If $\{e\}_\Theta(a) \simeq n$ then $\varphi_i(e,a) \simeq n$. Let e_i be the index for φ_i. Then $(e_i, e, a, n) \in \Gamma(\emptyset)$. Let $|e_i, e, a, n|_\Psi = |e,a,n|_\Theta$. For other tuples let $|e,a,r|_\Theta$ be the least τ such that $\tau > |e',a',r'|_\Psi$ for all immediate subcomputations (e',a',r') of (e,a,r). Then $(\Psi, ||_\Psi)$ is a computation theory.

<u>Lemma 16</u>: Let $\langle \Theta, ||_\Theta \rangle$ and $(\Psi, ||_\Psi)$ be as above. Then there is a Θ-computable mapping f such that for all e,a,m:

$$\{e\}_\Psi(a) \simeq m \iff \{f(e)\}_\Theta(p(a)) \simeq m;$$

if $\{e\}_\Psi(a) \simeq m$ and (e',a',m') is a subcomputation of (e,a,m) then $|\{f(e')\}_\Theta(p(a'))|_\Theta < |\{f(e)\}_\Theta(p(a))|_\Theta$.

There is a mapping g which is primitive recursive such that for all e,a,m:

$$\{e\}_\Theta(a) \simeq m \iff \{g(e)\}_\Psi(a) \simeq m;$$

if $\{e\}_\Theta(a) \simeq m$ and (e',a',m') is a subcomputation of (e,a,m) then $|\{g(e')\}_\Psi(a')|_\Psi < |\{g(e)\}_\Psi(a)|_\Psi$. ($p$ is the imbedding from A into $Tp(n)$ defined in § 3).

<u>Corollary</u>: Let X be a subset of $Tp(i)$, $i \leq n$.

i) X is Θ-semicomputable iff X is Ψ-semicomptable.

ii) X is Θ-computable iff X is Ψ-computable.

iii) X is weakly Θ-finite iff X is weakly Ψ-finite.

iv) X is strongly Θ-finite iff X is strongly Ψ-finite.

<u>Lemma 17</u>: A is strongly (weakly) Ψ-finite iff $Tp(n)$ is strongly (weakly) Θ-finite. S is strongly (weakly) Ψ-finite iff $Tp(n-1)$ is strongly (weakly) Θ-finite.

<u>Definition</u>: $||_\Psi$ is a <u>Ψ-norm</u> if the function $p(x,y)$ which compares the lengths of computations is Ψ-computable.

<u>Lemma 18</u>: If $||_\Psi$ is a Ψ-norm, then $||_\Theta$ is a Θ-norm. If $Tp(n)$ is weakly Θ-finite and $||_\Theta$ is a Θ-norm then $||_\Psi$ is a Ψ-norm.

<u>Definition</u>: Let $(\Psi, ||_\Psi)$ be a computation theory on $\mathfrak{A}$, and let $\mathcal{L}$ be a list of objects. <u>$\Psi \sim \mathcal{P}(\mathcal{L})$</u> if for all relations R on A: R is Ψ-semicomputable iff R is recursively enumerable in $\mathcal{L}$.

If $\langle \Theta, ||_\Theta \rangle$ is an abstract Kleene theory and $\underline{F}$ a list of objects of type $\leq n+2$ then the relation $\Theta \sim \underline{F}$ is defined in a similar way.

<u>Lemma 19</u>: Let $\langle \Theta, ||_\Theta \rangle$ be an abstract Kleene theory and let $(\Psi, ||_\Psi)$ be the associated computation theory. Then there is a list $\underline{F}$ such that $\Theta \sim \underline{F}$ iff there is a list $\mathcal{L}$ of total functions and functionals including the functions and functionals which describe the type structure, such that $\Psi \sim \mathcal{P}(\mathcal{L})$.

From the preceding lemmas in this chapter one can conclude: Given an abstract Kleene theory $\langle \Theta, ||_{\Theta} \rangle$ then there is a computation theory $(\Psi, ||_{\Psi})$ with almost the same properties as $\langle \Theta, ||_{\Theta} \rangle$. The converse of this is also true. If $(\Psi, ||_{\Psi})$ is a computation theory then there is an abstract Kleene theory $\langle \Theta, ||_{\Theta} \rangle$ with almost the same properties as $(\Psi, ||_{\Psi})$.

§ 8 NORMAL COMPUTATION THEORIES ON $\mathfrak{A}$

This chapter deals with normal computation theories. A normal computation theory is a generalization of recursion in normal lists. The main results are i) a characterization of which normal computation theories are equivalent to recursion in a normal list, and ii) a result which says that the computable relations on A and the semicomputable relations on S in a normal computation theory can always be obtained from a normal list. The latter result is a generalization of the " +1 " and " +2 " theorems.

A computation theory $(\Theta, ||_\Theta)$ on $\mathfrak{A}$ is <u>normal</u> if

i) the equality relation on S is Θ-computable,

ii) A is weakly Θ-finite and S is strongly Θ-finite,

iii) $(\Theta, ||_\Theta)$ is p-normal.

<u>Remark</u>: If F is a normal object of type $n+2$ then the computation theory obtained from F is normal.

Throughout this chapter $(\Theta, ||_\Theta)$ will be a normal computation theory. There are some interesting ordinals associated to Θ. Let X be a subset of A including the natural numbers. Let $\mathrm{Ord}(X) = \{ |\{e\}_\Theta(a)|_\Theta : \{e\}_\Theta(a)\downarrow$ and a is a list of elements from $X\}$.
Let

$$\varkappa^X = \sup \mathrm{Ord}(X)$$

$$\lambda^X = \text{the order type of } \mathrm{Ord}(X).$$

Particular cases: $X = N$, $N \cup \{x_1 \ldots x_m\}$, S, $S \cup \{x_1 \ldots x_m\}$, A. For these sets X the ordinal $\varkappa^X$ will be denoted by $\varkappa^0, \varkappa^a, \varkappa^S$, $\varkappa^{S,a}$, $\varkappa_\Theta$ respectively, where $a = x_1 \ldots x_m$. Similarly for λ^X.

Let B be a set and let $P \subseteq B \times B$. P is a prewellordering on B if P is a linear ordering, and there are no infinite descending chains in P. The domain of P ($\operatorname{dom} P$) is the set $\{x : \exists y((x,y) \in P \text{ or } (y,x) \in P)\}$. Let $|P|$ denote the length of the prewellordering.

Lemma 20: i) Let X be a subset of A including N. Then κ^X is the supremum of the lengths of the prewellorderings with domain $\subseteq A$ which are Θ-computable in elements from X. The supremum is not attained.

ii) If there are functions M_X, K_X, L_X such that X is closed under these functions, M_X is a pairing function on X with inverse functions K_X and L_X, and Θ-computable in elements from X, then λ^X is the supremum of the lengths of the prewellorderings with domain $\subseteq X$ which are Θ-computable in elements from X. The supremum is not attained.

Proof: Let P be a prewellordering which is Θ-computable in $a = x_1 \dots x_m$, where $x_1 \dots x_m \in X$, and $\operatorname{dom} P \subseteq A$. There is an index e_1 such that $\{e_1\}_\Theta(x,a) \simeq 0$ if $x \in \operatorname{dom} P$, and if $x \in \operatorname{dom} P$, then $|\{e_1\}_\Theta(y,a)| < |\{e_1\}_\Theta(x,a)|$ for all y below x in P. Hence the function $\rho : \operatorname{dom} P \to$ ordinals defined by $\rho(x) = |\{e_1\}_\Theta(x,a)|$ is orderpreserving. There is an index e_2 such that $\{e_2\}_\Theta(a)\downarrow$, and $|\{e_1\}_\Theta(x,a)|$ for all $x \in \operatorname{dom} P$. (Let $\{e_2\}_\Theta(a) \simeq E(f)$ where $f(x) \simeq \{e_1\}_\Theta(x,a)$ if $x \in \operatorname{dom} P$, $f(x) \simeq 1$ otherwise.)

To prove i) let P be as above. Then $|\{e_2\}_\Theta(a) \geq |P|$. Hence $\kappa^X > |P|$. If $\nu < \kappa^X$ choose an index e and a list a of elements from X such that $\{e\}_\Theta(a)\downarrow$ and $\nu < |\{e\}_\Theta(a)|$. Let P be the prewellordering defined by: $(x,y) \in P \iff$

$|x|_\Theta < |y|_\Theta < |\{e\}_\Theta(a)|$. Then P is Θ-computable in a, and $|P| = |\{e\}_\Theta(a)|$. Hence $\nu < |P|$.

To prove ii) let P be a prewellordering which is Θ-computable in a (a is a list from X), and $\operatorname{dom} P \subseteq X$. Then the set $\{|\{e_1\}_\Theta(x,a)| : x \in \operatorname{dom} P\}$ is a subset of $\mathrm{Ord}(X)$ and has order type $|P|$. $|\{e_2\}_\Theta(a)| \in \mathrm{Ord}(X)$, and $|\{e_1\}_\Theta(x,a)| < |\{e_2\}_\Theta(a)|$ for all $x \in \operatorname{dom} P$. The order type of $\mathrm{Ord}(X)$ is λ^X. Hence $|P| < \lambda^X$. Conversely let $\nu < \lambda^X$. Choose an index e and a list a of elements from X such that $\{e\}_\Theta(a)\downarrow$ and the order type of $\{\mu : \mu \in \mathrm{Ord}(X) \text{ and } \mu < |\{e\}_\Theta(a)|\} = \nu$. Let P' be the prewellordering defined by: $(x,y) \in P' \iff x \in \mathrm{Ord}(X)$ and $y \in \mathrm{Ord}(X)$ and $|x|_\Theta < |y|_\Theta < |\{e\}_\Theta(a)|$. Then P' is Θ-computable in a; and $|P'| = \nu$. $\mathrm{Dom}(P')$ is not necessarily a subset of X since $e',a' \in X$ does not imply that $\langle e',a'\rangle \in X$. We use the functions M_X, L_X, K_X to construct a prewellordering P such that $|P| = |P'|$, P is Θ-computable in elements from X and $\operatorname{dom} P \subseteq X$. This can be done as follows: Via the functions M_X, K_X, L_X one can code finite lists of elements in X as one element in X. Call the coding function $\langle\ \rangle_X$. From $\langle e',a'\rangle_X$ one can regain e',a' via decoding functions which are Θ-computable in elements in X. Let P be defined by: $(x,y) \in P \iff (x',y') \in P'$, where x' is obtained from x in the following way: If $x = \langle e',a'\rangle_X$ then $x' = \langle e',a'\rangle$. $\square$

<u>Lemma 21</u>: If X is one of the following sets: N, $N \cup \{x_1 \ldots x_n\}$, S, $S \cup \{x_1 \ldots x_n\}$ then $\lambda^X < \kappa^X < \kappa_\Theta$. If $a = (x_1 \ldots x_n)$ then $\lambda^a \leq \lambda^{S,a} < \kappa^a$.

Proof: Let X be one of the above sets. Then there are functions M_X, K_X, L_X which satisfies the hypothesis of ii) in lemma 20. If

P is a prewellordering with domain $\subseteq X$ then P can be regarded as an element in S_ω. There is an index e such that if x is a prewellordering with domain $\subseteq X$ then $\{e\}_\Theta(x)\downarrow$ and $|\{e\}_\Theta(x)| \geq$ the length of the prewellordering x. The relation "x is a prewellordering with domain $\subseteq X$" is Θ-computable in b, where $b = 0$ if $X = N$ or $X = S$, $b = a = (x_1 \ldots x_n)$ if $X = N \cup \{x_1 \ldots x_n\}$ or $X = S \cup \{x_1 \ldots x_n\}$. There is an index e' such that $\{e'\}_\Theta(b)\downarrow$ and $|\{e'\}_\Theta(b)| > |\{e\}_\Theta(x)|$ for all prewellorderings x with domain $\subseteq X$. By ii) of lemma 20 λ^X = supremum of the lengths of the prewellorderings with domain $\subseteq X$ which are Θ-computable in elements from X. Hence $\lambda^X < |\{e'\}_\Theta(b)|$, hence $\lambda^X < \varkappa^b \leq \varkappa^a \leq \varkappa^X$. This proves that $\lambda^X < \varkappa^X$, and $\lambda^{S,a} < \varkappa^a$ (let $X = S \cup \{x_1 \ldots x_n\}$). To prove that $\varkappa^X < \varkappa_\Theta$ regard tha set $\{\langle m,c\rangle : \{m\}_\Theta(c)\downarrow$, c is a list of elements from $X\}$. This set can be regarded as an element α of S_ω. An index e'' can be found such that $\{e''\}_\Theta(\alpha)\downarrow$, and $|\{e''\}_\Theta(\alpha)| > |\{m\}_\Theta(c)|$ for all $\langle m,c\rangle$ in the set. Hence $\varkappa^X \leq |\{e''\}_\Theta(\alpha)|$. Hence $\varkappa^X < \varkappa^\alpha \leq \varkappa_\Theta$. $\square$

Reflection: Harrington introduced the notion of reflection in recursion theory in his thesis [7]. In his exposition the notion is defined in the following way: An ordinal σ is a-reflecting if for each Σ_1-formula $\varphi(x)$ of a language $\mathcal{L}$: If $M_\sigma \models \varphi(a)$ then $M_{\varkappa^a} \models \omega(a)$. M_σ is a structure constructed from the set of computations of length less that σ. The interesting a-reflecting ordinals are those which are greater than $\varkappa^a$. Suppose σ is such an ordinal. Then the following is true: If a Σ_1-sentence $\varphi(a)$ is satisfied in the large model M_σ then it is satisfied in a smaller model. This is a reflecting property.

The notion of reflection can also be defined within the framework of this paper. An approach to the notion can be done by regarding the following problem: Let B be a Θ-semicomputable in a, nonempty subset of A. Is there a subset B' of B such that B' is nonempty and Θ-computable in a ?

This is not true for all B if the relations which are Θ-semicomputable, are not closed under existential quantifiers over A. This can be seen as follows: Let $P(x,a)$ be a Θ-semicomputable relation such that the relation $\exists x P(x,a)$ is not Θ-semicomputable. Let $B_a = \{x: P(x,a)\}$. Then B_a is Θ-semicomputable in a, and $B_a \neq \emptyset$ iff $R(a)$. To obtain a contradiction suppose that there is a nonempty subset B'_a of B_a which is Θ-computable in a for all a such that $B_a \neq \emptyset$. Since Θ admits selection operators for natural numbers there is a Θ-computable partial function φ such that if $B_a \neq \emptyset$ then $\varphi(a)$ is an index for the characteristic function of a nonempty subset of B_a. If $\varphi(a)\downarrow$ then $\lambda x\{\varphi(a)\}_\Theta(x,a)$ is the characteristic function of a nonempty subset of B_a. Hence $B_a \neq \emptyset \iff \varphi(a)\downarrow$. Hence $\exists x P(x,a) \iff \varphi(a)\downarrow$. This contradicts the fact that $\exists x P(x,a)$ is not Θ-semicomputable. Hence there is an a such that $B_a \neq \emptyset$, and there is no nonempty subset of B_a which is Θ-computable in a. $\square$

<u>Lemma 22</u>: Suppose $x \in B \iff \{e\}_\Theta(x,a)\downarrow$. Then there is a subset of B which is nonempty and Θ-computable in a iff $\exists x\, |\{e\}_\Theta(x,a)| < \kappa^a$.

Proof: Suppose B' is a nonempty subset of B, and B' is Θ-computable in a. There is an index e' such that $\{e'\}_\Theta(a)\downarrow$ iff $\forall x\,(x \in B' \Rightarrow \{e\}_\Theta(x,a)\downarrow)$, and $\{e'\}_\Theta(a)\downarrow \implies |\{e'\}_\Theta(a)| > |\{e\}_\Theta(x,a)|$ for all $x \in B'$. Now $\{e'\}_\Theta(a)\downarrow$, and $|\{e'\}_\Theta(a)| < \kappa^a$. Since

B' is nonempty $\exists x\,|\{e\}_\Theta(x,a)| < \varkappa^a$.

Suppose $\exists x\,|\{e\}_\Theta(x,a)| < \varkappa^a$. Then there is an e' such that $\{e'\}_\Theta(a)\downarrow$ and $\exists x\,|\{e\}_\Theta(x,a)| < |\{e'\}_\Theta(a)|$. Let B' be the subset of B defined by: $x \in B' \iff |\{e\}_\Theta(x,a) < |\{e'\}_\Theta(a)|$. Then B' is nonempty and Θ-computable in a. $\square$

<u>Lemma 23</u>: For all e, a: If $\exists x\,|\{e\}_\Theta(x,a)| < \varkappa^{S,a}$ then $\exists x\,|\{e\}_\Theta(x,a)| < \varkappa^a$.

<u>Proof</u>: Suppose $\exists x\,|\{e\}_\Theta(x,a)| < \varkappa^{S,a}$. Then

$\exists e' \in N\ \exists r \in S[\{e'\}_\Theta(r,a)\downarrow$ and $\exists x\,|\{e\}_\Theta(x,a)| < |\{e'\}_\Theta(r,a)|]$

There is an index e'' such that $\{e''\}_\Theta(a)\downarrow$ iff

$\exists e' \in N\ \exists r \in S[\{e'\}_\Theta(r,a)\downarrow$ and $\exists x\,|\{e\}_\Theta(x,a)| < |\{e'\}_\Theta(r,a)|]$,

and if $\{e''\}_\Theta(a)\downarrow$ then $|\{e''\}_\Theta(a)| > |\{e'\}_\Theta(r,a)|$ for some e', r. (e'' exists because Θ admits selection operators for numbers, and S is strongly Θ-finite. The quantifier $\exists x$ can be expressed by E since the relation $\lambda x\,|\{e\}_\Theta(x,a)| < |\{e'\}_\Theta(r,a)|$ is Θ-computable in a when $\{e'\}_\Theta(r,a)\downarrow$.) Now $\{e''\}_\Theta(a)\downarrow$, and $|\{e''\}_\Theta(a)| < \varkappa^a$. There is e', r such that $|\{e'\}_\Theta(r,a)| < |\{e''\}_\Theta(a)|$, and $\exists x\,|\{e\}_\Theta(x,a)| < |\{e'\}_\Theta(r,a)|$. Hence $\exists x\,|\{e\}_\Theta(x,a)| < \varkappa^a$. $\square$

Lemma 23 motivates the next definition.

<u>Definition</u>: Let σ be an ordinal $\leq \varkappa_\Theta$. Then σ is <u>a-reflecting</u> if for all $e \in N$: $\exists x\,|\{e\}_\Theta(x,a)| < \sigma \Rightarrow \exists x\,|\{e\}_\Theta(x,a)| < \varkappa^a$.

<u>Remark 1</u>: The a-reflecting ordinals are an initial segment of the ordinals. By lemma 23 $\varkappa^{S,a}$ is a-reflecting. In [21] this fact is called "simple reflection". If the Θ-semicomputable relations are not closed under existential quantifiers then by an earlier

discussion and lemma 22 $\varkappa_\Theta$ is not a-reflecting for all a.

If σ is a limit of a-reflecting ordinals then σ is a-reflecting. Hence there is a greatest a-reflecting ordinal. This ordinal is denoted by $\varkappa_r^a$.

If $S = Tp(0) \cup \ldots \cup Tp(n-1)$ where $n > 1$, and $(\Theta, ||_\Theta)$ is the normal computation theory obtained from a normal list then it can be proved that $\varkappa_\Theta$ is not a-reflecting for any a. $\varkappa^a < \varkappa^{S,a}$ because $n > 1$. Hence $\varkappa^a < \varkappa^{S,a} \leq \varkappa_r^a < \varkappa_\Theta$. In the following pages it will be proved that $\varkappa^{S,a} < \varkappa_r^a$ for all normal computation theories and all a.

<u>Remark 2</u>: There is an equivalent way of defining the a-reflecting ordinals. For $\tau < \varkappa_\Theta$ let $H_\tau = \{\langle e,a\rangle : |\langle e,a\rangle|_\Theta < \tau\}$. Let $\mathcal{F}$ be the class of formulas in the 1.order language which has a symbol for each of the following functions and predicates: N, s, S, M, K, L, $\lambda xi(x)_i$, $\langle \ldots \rangle_n$, S eq, lh. The language also has a unary predicate symbol X and a constant symbol for the number 0. $\sigma \leq \varkappa_\Theta$ is <u>a-reflecting</u> if for each $\varphi(x,X)$ in $\mathcal{F}$: $\exists \tau < \sigma \quad \varphi(a,H_\tau) \Longrightarrow \exists \tau < \varkappa^a \quad \varphi(a,H_\tau)$. This definition will not be used in this paper.

<u>Lemma 24</u>: Suppose $x \in B \iff \{e\}_\Theta(x,a)\downarrow$. Then i), ii) and iii) are equivalent.

i) There is a subset B' of B which is nonempty and Θ-computable in a.

ii) $\exists x \, |\{e\}_\Theta(x,a)| < \varkappa^a$

iii) $\exists x \, |\{e\}_\Theta(x,a)| < \varkappa_r^a$.

The next lemma is a characterization of the subsets of A which are strongly Θ-finite.

Lemma 25: Let $B \subseteq A$ be Θ-computable. Then i), ii), and iii) are equivalent.

i) B is strongly Θ-finite,

ii) for all e,a: $\exists x \in B\, \{e\}_\Theta(x,a)\downarrow \implies \exists x \in B\, |\{e\}_\Theta(x,a)| < \varkappa^a$,

iii) for all a and all subsets C of B: if C is nonempty and Θ-semicomputable in a then there is a nonempty subset C' of C which is Θ-computable in a.

Proof: i) $\Rightarrow$ ii). Let $\mathcal{F}_B$ be the partial functional defined by:

$$\mathcal{F}_B(\varphi) \simeq \begin{cases} 0 & \text{if } \exists x \in B \quad \varphi(x) \simeq 0 \\ 1 & \text{if } \forall x \in B \quad \exists y \neq 0 \quad \varphi(x) \simeq y \end{cases}$$

By definition B is strongly Θ-finite iff the functional $\mathcal{F}_B$ is weakly Θ-computable. Suppose B is strongly Θ-finite and $\exists x \in B\, \{e\}_\Theta(x,a)\downarrow$. Choose an index $\hat{e}$ such that $\{\hat{e}\}_\Theta(x,a) \simeq t(\{e\}_\Theta(x,a))$, where t is the constant function with value 0, and $|\{\hat{e}\}_\Theta(x,a)| > |\{e\}_\Theta(x,a)|$ when $\{e\}_\Theta(x,a)\downarrow$. Then $\mathcal{F}_B(\varphi) \simeq 0$ where $\varphi = \lambda x \{\hat{e}\}_\Theta(x,a)$. Since $\mathcal{F}_B$ is weakly Θ-computable there is an index e' and a subfunction φ' of φ such that $\mathcal{F}_B(\varphi') \simeq 0$, $\{e'\}_\Theta(a) \simeq 0$, $|\{e'\}_\Theta(a)| > |\{\hat{e}\}_\Theta(x,a)|$ for all $x \in \operatorname{dom} \varphi'$. Since $\mathcal{F}_B(\varphi') \simeq 0$ there is an x in B such that $\varphi'(x) \simeq 0$. Hence $\inf\{|\{\hat{e}\}_\Theta(x,a)| : x \in B \text{ and } \varphi(x) \simeq 0\} < |\{e'\}_\Theta(a)| < \varkappa^a$.

To prove ii) $\Rightarrow$ i) suppose $\exists x \in B\, \{e\}_\Theta(x,a)\downarrow \implies \exists x \in B\, |\{e\}_\Theta(x,a)| < \varkappa^a$. As a relation of e, a the relation $\exists x \in B\, |\{e\}_\Theta(x,a)| < \varkappa^a$ is Θ-semicomputable. For $\exists x \in B\, |\{e\}_\Theta(x,a)| < \varkappa^a \iff \exists n(\{n\}_\Theta(a)\downarrow \text{ and } \exists x \in B\, |\{e\}_\Theta(x,a)| < |\{n\}_\Theta(a)|)$. B is weakly Θ-finite since B is Θ-computable and the computation theory is normal. An index for the relation "$\{n\}_\Theta(a)\downarrow$ and $\exists x \in B\, |\{e\}_\Theta(x,a)| < |\{n\}_\Theta(a)|$" can be constructed from an index for the functional which proves that B is weakly Θ-finite, and an index for the function which compares lengths of computations. Since Θ admits selection operators for natural numbers, there is a partial Θ-computable function $\varphi(e,a)$ such that if for some n $\{n\}_\Theta(a)\downarrow$ and $\exists x \in B\ |\{e\}_\Theta(x,a)| < |\{n\}_\Theta(a)|$ then $\varphi(e,a)$ is such an n. Conversely if $\varphi(e,a)\downarrow$ then $\varphi(e,a)$ is such an n. Construct an index e' such that $\{e'\}_\Theta(e,a) \simeq \{\varphi(e,a)\}_\Theta(a)$, and $|\{e'\}_\Theta(e,a)| > |\{\varphi(e,a)\}_\Theta(a)|$ when $\varphi(e,a)\downarrow$. Then $\exists x \in B\ \{e\}_\Theta(x,a)\downarrow$ iff $\{e'\}_\Theta(e,a)\downarrow$, and if $\exists x \in B\ \{e\}_\Theta(x,a)\downarrow$ then $\exists x \in B\ |\{e\}_\Theta(x,a)| < |\{e'\}_\Theta(e,a)|$. Construct e'' from e' such that $\exists x \in B\ \{e\}_\Theta(x,a) \simeq 0$ iff $\{e''\}_\Theta(e,a)\downarrow$, and if $\{e''\}_\Theta(e,a)\downarrow$ then $\exists x \in B\ (\{e\}_\Theta(x,a) \simeq 0$ and $|\{e\}_\Theta(x,a)| < |\{e''\}_\Theta(e,a)|)$. Since B is weakly Θ-finite there is an index e''' such that $\forall x \in B\ \exists y \neq 0\ \{e\}_\Theta(x,a) \simeq y$ iff $\{e'''\}_\Theta(e,a) \simeq 1$, and if $\{e'''\}_\Theta(e,a) \simeq 1$ then $|\{e\}_\Theta(x,a)| < |\{e''\}_\Theta(e,a)|$ for all $x \in B$. An index for $\mathscr{F}_B$ can be found from e'' and e'''. Hence B is strongly Θ-finite.

ii) $\Rightarrow$ iii). Let C be a nonempty subset of B which is Θ-semicomputable in a. Choose e such that $x \in C \iff \{e\}_\Theta(x,a)\downarrow$. Since C is nonempty $\exists x \in B\ \{e\}_\Theta(x,a)\downarrow$. By ii) $\exists x \in B\ |\{e\}_\Theta(x,a)| < \kappa^a$. Choose n such that $\{n\}_\Theta(a)\downarrow$ and $x \in B\ |\{e\}_\Theta(x,a)| < |\{n\}_\Theta(a)|$. Let $C' = \{x : |\{e\}_\Theta(x,a)| < |\{n\}_\Theta(a)|\}$. Then C' is a nonempty subset of C, and C' is Θ-computable in a.

iii) $\Rightarrow$ ii). Suppose $x \in B\ \{e\}_\Theta(x,a)\downarrow$. Let $C = \{x : x \in B$ and $\{e\}_\Theta(x,a)\downarrow\}$. Then C is a nonempty subset of B which is Θ-semicomputable in a. By iii) there is a nonempty subset C' of C which is Θ-computable in a. An index e' for the relation "$\forall x\,(x \in C' \Rightarrow \{e\}_\Theta(x,a)\downarrow)$" can be found such that $\{e'\}_\Theta(a)\downarrow$, and $|\{e\}_\Theta(x,a)| < |\{e'\}_\Theta(a)|$ for all $x \in C'$. Hence

$\exists x \in B \ |\{e\}_\Theta(x,a)| < |\{e'\}_\Theta(a)| < \varkappa^a$. This proves ii). $\square$

Let a be a fixed finite list of elements in A. Let $P = \{\langle e,b^-\rangle : \{e\}_\Theta(b)\downarrow$ and b is a list of elements from S and the list a. b^- is the list obtained by removing the elements of a from b.$\}$

P is a subset of S, hence an element of ${}^S\omega$.

P is a complete Θ-semicomputable in a, subset of S, i.e. all other subsets of S which are Θ-semicomputable in a can be reduced to P by a Θ-computable one-one mapping.

<u>Theorem 8</u>: $\varkappa^{S,P,a}$ is a-reflecting.

($\varkappa^{S,P,a} = \varkappa^X$ where $X = S \cup \{P\} \cup \{x : x$ is in the list $a\}$.)

In [21] this result is called "further reflection". To prove theorem 8 we first prove three propositions.

Let pwo_S denote the set of prewellorderings with domain $\subseteq S$. If $pwo_S(x)$ let $|x|$ denote the length of the prewellordering. If $y \in dom(x)$ let $|y|_x$ denote the length of that part of the prewellordering which is below y. There is a natural prewellordering with domain P and length $\lambda^{S,a}$ defined by: $(\langle e,b^-\rangle,\langle e',b'^-\rangle)$ is in the prewellordering if $|\{e\}_\Theta(b)|_\Theta < |\{e'\}_\Theta(b')|_\Theta$ for $\langle e,b^-\rangle,\langle e',b'^-\rangle \in P$. If x is in this prewellordering let $|x|_P$ denote the length of that part of the prewellordering which is below x.

<u>Proposition 1</u>: The relation "$pwo_S(x)$ and $|x| < \lambda^{S,a}$" is Θ-semicomputable as a relation of x,a. If $|x| \geq \lambda^{S,a}$ then P is Θ-computable in x,a and possibly an element in S.

Proof: The relation $pwo_S(x)$ is Θ-computable, for $pwo_S(x) \Leftrightarrow x$

is a total transitive ordering and there are no infinite descending chains in the ordering. Since the domain of the ordering is a subset of S an infinite descending chain in the ordering can be regarded as an element of $^{S}\omega$. Since $^{S}\omega$ is weakly finite the expression "there are no infinite descending chains" is Θ-computable.

There is a Θ- computable partial function φ such that $\varphi(x,y,z,a) \simeq 0$ iff $\mathrm{pwo}_S(x)$ and $z \in \mathrm{dom}\, x$, $y \in P$, $|z|_x \leq |y|_P$. (To prove that such a φ exists use the first recursion theorem.) Then $\mathrm{pwo}_S(x)$ and $|x| < \lambda^{S,a} \Longleftrightarrow \mathrm{pwo}_S(x)$ and $\exists y\, [y \in P$ and $\forall z \in \mathrm{dom}\, x \;\; \varphi(x,y,z,a) \simeq 0]$.
Since the quantifier $\exists y$ ranges over S the right hand side of the equivalence is Θ-semicomputable.

By another application of the first recursion theorem one can prove that there is a Θ-computable partial function ψ such that

$$\psi(x,y,z,a) \simeq 0 \text{ if } \mathrm{pwo}_S(x),\ z \in \mathrm{dom}\, x,\ |z|_x \leq |y|_P$$
$$\simeq 1 \text{ if } \mathrm{pwo}_S(x),\ z \in \mathrm{dom}\, x,\ y \in P \text{ and } |z|_x > |y|_P$$

(Convention: $|y|_P = \lambda^{S,a}$ if $y \notin P$.)

Suppose $|x| \geq \lambda^{S,a}$. In a simple way one can define a prewellordering x' from x such that $|x'| = |x| + 1$. Then $|x'| > \lambda^{S,a}$. There is an r in $\mathrm{dom}(x')$ such that $|r|_{x'} = \lambda^{S,a}$. $\lambda y\, (1 - \psi(x',y,r,a))$ is the characteristic function of P. Hence P is Θ-computable in x,a,r. $\square$

Proposition 2: If P is Θ-computable in a,x and elements from S then $\kappa^{S,P,a} \leq \kappa^{S,x,a}$.

Proof: Let $\nu < \kappa^{S,P,a}$ be arbitrary. Then for some e,b: $\{e\}_\Theta(b)\downarrow$ and $\nu < |\{e\}_\Theta(b)|$, where the list b can contain P and elements from a and S. Since P is Θ-computable in a,x

and elements $r_1 \ldots r_k \in S$ there is an index e' such that $\lambda y\, \{e'\}_{\Theta}(y,x,a,r_1 \ldots r_k)$ is the characteristic function of P. Substitute this function for P in $\{e\}_{\Theta}(b)$ and obtain a computation $\{e''\}_{\Theta}(c)$ such that $\{e''\}_{\Theta}(c)\downarrow$ and $|\{e\}_{\Theta}(b)| < |\{e''\}_{\Theta}(c)|$, where the list c can contain x and elements from a and S. $|\{e''\}_{\Theta}(c)| < \varkappa^{S,x,a}$. Hence $\nu < \varkappa^{S,x,a}$, and $\varkappa^{S,P,a} \leq \varkappa^{S,x,a}$. $\square$

Proposition 3: There is an index $\hat{e}$ such that if $\exists y|\{e\}_{\Theta}(y,a)| < \varkappa^{S,a,x}$ then $\{\hat{e}\}_{\Theta}(e,a,x)\downarrow$ and $\exists y|\{e\}_{\Theta}(y,a)| < |\{\hat{e}\}_{\Theta}(e,a,x)|$.

Proof: $\exists y|\{e\}_{\Theta}(y,a)| < \varkappa^{S,a,x} \iff \exists e'\ r \in S[\{e'\}_{\Theta}(r,x,a)\downarrow$ and $\exists y\,|\{e\}_{\Theta}(y,a)| \leq |\{e'\}_{\Theta}(r,x,a)|]$. Let $C = \{\langle e',r\rangle : \{e'\}_{\Theta}(r,x,a)\downarrow$ and $\exists y\,|\{e\}_{\Theta}(y,a)| \leq |\{e'\}_{\Theta}(r,x,a)|\}$. Then C is a subset of S which is Θ-semicomputable in a,x. Suppose $\exists y\,|\{e\}_{\Theta}(y,a)| < \varkappa^{S,a,x}$. Then C is nonempty. S is strongly Θ-finite since $(\Theta, ||_{\Theta})$ is normal. By lemma 25 there is a nonempty subset C' of C which is Θ-computable in a,x. An index for C' can be found uniformly from an index for C. There is an index $\hat{e}$ for the relation "$\forall e',r\,(\langle e',r\rangle \in C' \Rightarrow \{e'\}_{\Theta}(r,x,a)\downarrow)$" such that $\{\hat{e}\}_{\Theta}(e,a,x)\downarrow$ and $|\{e'\}_{\Theta}(r,x,a)| < |\{\hat{e}\}_{\Theta}(e,a,x)|$ for all $\langle e',r\rangle \in C'$. Since $\exists y\,|\{e\}_{\Theta}(y,a)| < |\{e'\}_{\Theta}(r,x,a)|$ for all $\langle e',r\rangle \in C'$, $\exists y\,|\{e\}_{\Theta}(y,a)| < |\{\hat{e}\}_{\Theta}(e,a,x)|$. $\square$

Proof of theorem 8: Suppose $\exists y\,|\{e\}_{\Theta}(y,a)| < \varkappa^{S,P,a}$. If $\mathrm{pwo}_S(x)$ then either $|x| < \lambda^{S,a}$ or $|x| \geq \lambda^{S,a}$. In the latter case P is Θ-computable in a,x and elements from S by proposition 1. By proposition 2 $\varkappa^{S,P,a} \leq \varkappa^{S,x,a}$, hence $\exists y\,|\{e\}_{\Theta}(y,a) < \varkappa^{S,x,a}$. Thus for all x: $\mathrm{pwo}_S(x) \Rightarrow |x| < \lambda^{S,a}$ or $\exists y\,|\{e\}_{\Theta}(y,a)| < \varkappa^{S,x,a}$. By proposition 1 the relation "$\mathrm{pwo}_S(x)$ and $|x| < \lambda^{S,a}$"

is Θ-semicomputable. Let e_1 be an index for this relation. By proposition 3 there is an index $\hat{e}$ such that if $\exists y|\{e\}_\Theta(y,a)| < \kappa^{S,x,a}$ then $\{\hat{e}\}_\Theta(e,a,x)\downarrow$ and $\exists y\,|\{e\}_\Theta(y,a)| < |\{\hat{e}\}_\Theta(e,a,x)|$. There is an index f such that $\{f\}_\Theta(e,a)\downarrow$ if for all x: $pwo_S(x) \implies |x| < \lambda^{S,a}$ or $\exists y\,|\{e\}_\Theta(y,a)| < \kappa^{S,x,a}$, and in this case $|\{f\}_\Theta(e,a)| > \inf\{|\{e_1\}_\Theta(x,a)|,\ |\{\hat{e}\}_\Theta(e,a,x)|\}$ for all x such that $pwo_S(x)$. Now $\{f\}_\Theta(e,a)\downarrow$. Choose x such that $pwo_S(x)$ and $|x| \geq \lambda^{S,a}$. Then $\{e_1\}_\Theta(x,a)\uparrow$, and $\exists y\,|\{e\}_\Theta(y,a)| < \kappa^{S,x,a}$. Hence $\exists y\,|\{e\}_\Theta(y,a)| < |\{\hat{e}\}_\Theta(e,a,x)| < |\{f\}_\Theta(e,a)| < \kappa^a$. $\square$

<u>Remark</u>: $\kappa^{S,a} < \kappa^{P,a} \leq \kappa^{S,P,a}$. To prove that $\kappa^{S,a} < \kappa^{P,a}$ let e be an index for the expression $\forall\langle e',b^-\rangle\ (\langle e',b^-\rangle \Rightarrow \{e'\}_\Theta(b)\downarrow)$ such that $\{e\}_\Theta(P,a)\downarrow$ and $|\{e'\}_\Theta(b)| < |\{e\}_\Theta(P,a)|$ for all $\langle e',b^-\rangle$ in P. Since $\kappa^{S,a} = \sup\{|\{e'\}_\Theta(b)| : \langle e',b^-\rangle \in P\}$, $\kappa^{S,a} \leq |\{e\}_\Theta(P,a)|$. Hence $\kappa^{S,a} < \kappa^{P,a}$ since $|\{e\}_\Theta(P,a)| < \kappa^{P,a}$.

<u>Corollary</u>: Suppose that B is a set of subsets of S such that B is Θ-semicomputable in a, and B contains a subset of S which is nonempty and Θ-semicomputable in a. Then B contains a subset of S which is nonempty and Θ-computable in a.

Proof: Let $x \in B \iff \{e\}_\Theta(x,a)\downarrow$. Let $C \in B$ be Θ-semicomputable in a with index e', i.e. $r \in C \iff \{e'\}_\Theta(r,a)\downarrow$. For $\sigma < \kappa_\Theta$ let C_σ be defined by: $C_\sigma = \{r : |\{e'\}_\Theta(r,a)| < \sigma\}$. If $\sigma \geq \kappa^{S,a}$ then $C_\sigma = C$. If y is a convergent computation and $|y|_\Theta = \sigma$ then C_σ is Θ-computable in a,y, uniformly. Let g be an index such that $\{g\}_\Theta(y,a)\downarrow$ iff $C_{|y|_\Theta} \in B$. By the remark $\kappa^{S,a} < \kappa^{P,a}$. Choose $y = \langle f,P,a\rangle$ such that $\{f\}_\Theta(P,a)\downarrow$ and $\kappa^{S,a} \leq |\{f\}_\Theta(P,a|$. Then $C_{|y|_\Theta} = C$, $C_{|y|_\Theta} \in B$, $\{g\}_\Theta(y,a)\downarrow$, and

$|\{g\}_\Theta(y,a)| < \varkappa^{P,a}$. By theorem 8 $\varkappa^{S,P,a}$ is reflecting. Hence $\varkappa^{P,a}$ is reflecting, and $\exists y\, |\{g\}_\Theta(y,a)| < \varkappa^a$. Choose $h \in \omega$ such that $\{h\}_\Theta(a)\downarrow$ and $\exists y\, |\{g\}_\Theta(y,a)| < |\{h\}_\Theta(a)|$. The set $D = \{y : |\{g\}_\Theta(y,a)| < |\{h\}_\Theta(a)|$ and $|y|_\Theta$ is minimal$\}$ is Θ-computable in a. Let $\tau = |y|_\Theta$ where $y \in D$. Then $C_\tau \in B$ because $\{g\}_\Theta(y,a)\downarrow$ when $y \in D$. $r \in C_\tau \iff \forall y\, (y \in D \Rightarrow |\{e'\}_\Theta(r,a)| < |y|_\Theta)$. The relation inside the brackets is Θ-computable in a. Hence C_τ is Θ-computable in a. $\square$

Notations:

$sc(\Theta) \quad = \{X : X \subseteq A$ and X is Θ-computable$\}$

$sc(\Theta,a) \quad = \{X : X \subseteq A$ and X is Θ-computable in $a\}$

$S - sc(\Theta,a) = \{X : X \subseteq S$ and X is Θ-computable in $a\}$

$en(\Theta) \quad = \{X : X \subseteq A$ and X is Θ-semicomputable$\}$

$S - en(\Theta) \quad = \{X : X \subseteq S$ and X is Θ-semicomputable$\}$

If $\mathcal{L}$ is a normal list we write $en(\mathcal{L})$, $sc(\mathcal{L})$ instead of $en(\Theta)$, $sc(\Theta)$, where $(\Theta, ||_\Theta)$ is the normal computation theory obtained from $\mathcal{L}$.

If $F \in Tp(m)$ then for $n > 0$:

$n - sc(F) \quad = \{X : X \subseteq Tp(n-1)$ and X is recursive in $F\}$

$n - en(F) \quad = \{X : X \subseteq Tp(n-1)$ and X is recursively enumerable in $F\}$

In the rest of this chapter the following problem will be considered. Let $(\Theta, ||_\Theta)$ be a normal computation theory. Is there a list $\mathcal{L}$ such that $(\Theta, ||_\Theta)$ is similar to recursion in $\mathcal{L}$? By "$(\Theta, ||_\Theta)$ is similar to recursion in $\mathcal{L}$ " we will mean one of the following two statements: $en(\Theta) = en(\mathcal{L})$; for all partial functions φ : φ is Θ-computable iff φ is recursive in $\mathcal{L}$. (When $S = \omega$

these two statements are equivalent.)

We have the following result from recursion in higher types: If F is a normal object of type $> n+2$ then there is no normal object G of type $n+2$ such that $n+1-en(F) = n+1-en(G)$. The case $n = 0$ is proved by Moschovakis in [17]. When $n > 0$ then $n+1-en(F)$ is closed under the quantifier $\exists x \in Tp(n)$, and $n+1-en(G)$ is not for any $G \in Tp(n+2)$. Hence $n+1-en(F) \neq n+1-en(G)$.

This result can be translated to the setting of this paper. Let F be as above. Let $S = Tp(0) \cup \ldots \cup Tp(n-1)$, let $\mathfrak{A}$ be the computation domain with subindividuals S, and let $(\Theta, ||_\Theta)$ be the normal computation theory on $\mathfrak{A}$ obtained from F as described in § 3. Then there is no normal list $\mathcal{L}$ such that any of the two statements above is true.

The following two results are proved for recursion on the types: If F is a normal object to type $> n+2$ then there is a normal object G of type $n+2$ such that $n+1-sc(F) = n+1-sc(G)$. If F is a normal object to type $> n+2$ where $n > 0$ then there is a normal object G of type $n+2$ such that $n-en(F) = n-en(G)$. The first result is called the " +1 theorem " and is proved by Sacks in [20] and [21]. The other result is called the " +2 theorem " and is proved by Harrington in [7].

Theorem 9 is a generalization of these two theorems. Both theorems are corollaries of theorem 9 when $n > 0$: If F is a normal object of type $> n+2$ where $n > 0$, let $(\Theta, ||_\Theta)$ be the normal computation theory obtained from F as mentioned above. By theorem 9 there is a normal list $\mathcal{L}$ such that $S-en(\Theta) = S-en(\mathcal{L})$, $sc(\Theta,r) = sc(\Theta,r)$ for all $r \in S$. Hence there is a normal object G of type $n+2$ such that $n-en(F) = n-en(G)$ and

$n{+}1 - sc(F,x) = n{+}1 - sc(G,x)$ for all $x \in Tp(n-1)$.

<u>Theorem 9</u>: Let $(\Theta, ||_{\Theta})$ be normal. Then there is a normal list $\mathcal{L}$ such that $S\text{-}en(\Theta) = S\text{-}en(\mathcal{L})$, and $sc(\Theta,r) = sc(\mathcal{L},r)$ for all $r \in S$.

Proof: Let $P = \{\langle e,a\rangle : \{e\}_{\Theta}(a)\downarrow$ and a is a list of objects from $S\}$. An ordinal η is <u>Θ-subconstructive</u> if $\eta = |\{e\}_{\Theta}(a)|_{\Theta}$ for some $\langle e,a\rangle$ in P. The order type of the Θ-subconstructive ordinals is $\lambda = \lambda^S$. For $\nu < \lambda$ let η_ν be the ν-th Θ-subconstructive ordinal.

The list $\mathcal{L}$ will contain three objects: the functionals E and G and the equality relation on S. G is constructed in stages. If τ is an ordinal then G_τ is a partial functional. The domain of G_τ ($\operatorname{dom} G_\tau$) contains that part of $\operatorname{dom} G$ which is needed to generate H_τ, where $H_\tau = \{\langle e,a\rangle : |\{e\}^{\mathcal{L}}(a)| < \tau\}$. If $\tau < \tau'$ then $G_\tau \subseteq G_{\tau'}$, i.e. if $f \in \operatorname{dom} G_\tau$ then $f \in \operatorname{dom} G_{\tau'}$ and $G_\tau(f) = G_{\tau'}(f)$. Finally G is defined by:

$$G(f) = \begin{cases} (\bigcup_\tau G_\tau)(f) & \text{if } f \in \bigcup_\tau \operatorname{dom} G_\tau \\ 0 & \text{otherwise} \end{cases}$$

Let σ_ν be the ν-th $\mathcal{L}$-subconstructive ordinal.

Suppose G_τ, H_τ are defined.

$H_{\tau+1} = \{\langle e,a\rangle :$ all immediate subcomputations of $\{e\}^{\mathcal{L}}(a)$ are in $H_\tau\}$. If $\forall x(\{e\}^{\mathcal{L}}(x,a)\downarrow$ and $|\{e\}^{\mathcal{L}}(x,a)| < \tau)$ then $\langle\langle 16,n,e,2\rangle,a\rangle \in H_{\tau+1}$ $(\{\langle 16,n,e,2\rangle\}^{\mathcal{L}}(a) \simeq G(\lambda x\{e\}^{\mathcal{L}}(x,a)))$. Let $f = \lambda x\{e\}^{\mathcal{L}}(x,a)$. If f is not already in the domain of G_τ we must define $G(f)$ in this stage. Hence we let $f \in \operatorname{dom} G_{\tau+1}$, and $G_{\tau+1}(f) = 0$ or 1. $G_{\tau+1}$ is said to be the trivial extension of G_τ if $\operatorname{dom} G_{\tau+1} = \operatorname{dom} G_\tau \cup \{f : f$ is as above$\}$, and $G_{\tau+1}(f) = 0$ when

when $f \in \operatorname{dom} G_{\tau+1} - \operatorname{dom} G_\tau$. In the construction of $G_{\tau+1}$ we let $G_{\tau+1}(f) = 0$ when f is as above and $f \notin \operatorname{dom} G_\tau$, if not otherwise mentioned.

To obtain $S\text{-}\mathrm{en}(\Theta) = S\text{-}\mathrm{en}(\mathcal{L})$ information about Θ must be brought into the construction of G. This is done at stages which are $\mathcal{L}$-subconstructive. Suppose we have constructed G_μ for all $\mu < \tau$, and $\tau = \sigma_\nu$ for a $\nu < \lambda$. Regard the set $\{x: |x|_\Theta \leq \eta_\nu\}$. For each x in this set we take a function f $(= f_{xy})$ such that $f \notin \operatorname{dom} G_\mu$ when $\mu < \tau$, and let $G_\tau(f) = 1$. From f and G_τ one can regain the information that x is in the above set. f is also recursive in $\mathcal{L}, x, y$, where y is any $\mathcal{L}$-computation of length τ. The existence of such a set of functions is proved in the following proposition.

<u>Proposition 1</u>: Let $\mathcal{L}$ be any normal list. Let y be a convergent $\mathcal{L}$-computation with length τ. For each $x \in A$ there is a total function f_{xy} such that f_{xy} is recursive in $\mathcal{L}, x, y$, and if $x \neq x'$ then $f_{xy} \neq f_{x'y}$. If $f_{xy} = \lambda t\, \{e\}^{\mathcal{L}}(t,a)$ for some e, a then $\tau \leq |\{e\}^{\mathcal{L}}(t,a)|$ for some $t \in A$.

Proof: Let τ^+ be the least limit ordinal $\geq \tau$. The set of $\mathcal{L}$-computations with length $< \tau^+$ is recursive in $\mathcal{L}, y$. Let f_y be the function defined by: $f_y(u) = 0$ if u is not a sequence $\langle e,a\rangle$. If $u = \langle e,a\rangle$ then see if $|\{e\}^{\mathcal{L}}(t,a)| < \tau^+$ for all t. If not let $f_y(u) = 0$. If true let $f_y(u)$ be something different from $\{e\}^{\mathcal{L}}(\langle e,a\rangle,a)$. Let for instance $f_y(\langle e,a\rangle) = 1$ if $\{e\}^{\mathcal{L}}(\langle e,a\rangle,a) \simeq 0$, $f_y(\langle e,a\rangle) = 0$ otherwise. f_y is recursive in $\mathcal{L}, y$. If $f = \lambda t\, \{e\}^{\mathcal{L}}(t,a)$ is a total function such that $|\{e\}^{\mathcal{L}}(t,a)| < \tau^+$ for all t then f_y and f have different values for $t = \langle e,a\rangle$.

Hence $f \neq f_y$. Let

$$\begin{aligned} f_{xy}(t) &= \langle f_y(t), x, 0\rangle \quad \text{if} \quad x \in S \\ &= \langle f_y(t), x(t), 1\rangle \quad \text{if} \quad x \in {}^{S}\omega,\ t \in S \\ &= f_y(t) \quad \text{if} \quad x \in {}^{S}\omega,\ t \in {}^{S}\omega \end{aligned}$$

Then f_{xy} is recursive in $\mathscr{L}, x, y$, uniformly in x, y. If $x \neq x'$ then $f_{xy} \neq f_{x'y}$. If $f_{xy} = \lambda t\, \{e\}^{\mathscr{L}}(t,a)$ then $\tau \leq |\{e\}^{\mathscr{L}}(t,a)|$ for some t. For suppose $|\{e\}^{\mathscr{L}}(t,a)| < \tau$ for all t. The function f_y can easily be regained from f_{xy}. In fact there is an index e' such that $f_y = \lambda t\, \{e'\}^{\mathscr{L}}(t,a)$ and $|\{e'\}^{\mathscr{L}}(t,a)| < |\{e\}^{\mathscr{L}}(t,a)| + \omega$ for all t. Hence $|\{e'\}^{\mathscr{L}}(t,a)| < \tau^+$ for all t. Hence $f_y(\langle e', a\rangle \neq \{e'\}^{\mathscr{L}}(\langle e', a\rangle, a)$, a contradiction. $\square$

Construction of G_τ :

Suppose G_μ, H_μ are defined for all $\mu < \tau$. We define G_τ in two cases:

Case 1: <u>There is an ordinal $\nu < \lambda$ such that ν is the order type of the ordinals $< \tau$ which are $\mathscr{L}$ - subconstructive</u>

This case is divided into two:

I: <u>τ is $\mathscr{L}$-subconstructive</u> (i.e. $\tau = \sigma_\nu$).

Let $G_\tau(f_{xy}) = 1$ for all x, y such that x is a Θ-computation of length $\leq \eta_\nu$, y is an $\mathscr{L}$-computation of length τ.

II: <u>τ is not $\mathscr{L}$-subconstructive</u> (i.e. $\tau < \sigma_\nu$).

Let $\epsilon = \eta_\nu - \sup\{\eta_\rho : \rho < \nu\}$. The definition of G_τ depends on the answer to the following question: Is there an ordinal π such that $\tau < \pi \leq \tau + \epsilon$ and π is $\mathscr{L}^0$-subconstructive?

Yes: $G_\tau = \bigcup_{\mu<\tau} G_\mu$ if $\lim \tau$, G_τ is the trivial extension of G_μ if $\tau = \mu + 1$.

No : $G_\tau(f_{xy}) = 1$ when x, y are as in I.

Case II: <u>Otherwise</u>

$G_\tau = \bigcup_{\mu<\tau} G_\mu$ if $\lim \tau$, G_τ is the trivial extension of G_μ if $\tau = \mu + 1$.

<u>Remark</u>: One can decide whether or not τ is $\mathcal{L}$-subconstructive before G_τ is defined. For H_μ is defined when $\mu < \tau$. If τ is a limit ordinal then τ is $\mathcal{L}$-subconstructive iff there are $e \in N$ and a list a of objects from S such that $\{e\}^{\mathcal{L}}(t,a)\downarrow$ for all t, and $\sup\{|\{e\}^{\mathcal{L}}(t,a)| + 1 : t \in A\} = \tau$ iff there are e and a as above such that $\langle e,t,a\rangle \in \bigcup_{\mu<\tau} H_\mu$ for all t, and if $\mu < \tau$ then for some t $\langle e,t,a\rangle \notin H_\mu$. If τ is a successor ordinal there is a similar test to decide from $\bigcup_{\mu<\tau} H_\mu$ whether or not τ is $\mathcal{L}$-subconstructive.

The list $\mathcal{L}^o$ in subcase II of case 1 consists of the two functionals G^o, E and the equality relation on S. Hence $\mathcal{L}^o$ is normal. G^o is defined by:

$$G^o(f) = (\bigcup_{\mu<\tau} G_\mu)(f) \quad \text{if } \exists\mu < \tau \quad f \in \operatorname{dom} G_\mu ,$$

$$G^o(f) = 0 \quad \text{othe wise.}$$

The functions f_{xy} are as in proposition 1 with $\mathcal{L} = \mathcal{L}^o$. Then f_{xy} is recursive in $\mathcal{L}^o,x,y$, hence in $\mathcal{L},x,y$ since G^o is recursive in $\mathcal{L},y$. Also $f_{xy} \notin \operatorname{dom} G_\mu$ when $\mu < \tau$.

This defines G, and we prove that $\mathcal{L}$ has the desired properties.

<u>Proposition 2</u>: The order type of the $\mathcal{L}$-subconstructive ordinals is at least λ.

Proof: Suppose not. Let $\nu < \lambda$ be the order type of the $\mathcal{L}$-subconstructive ordinals. Let $\tau = \sup\{\sigma_\rho : \rho < \nu\}$. τ is the supremum of the $\mathcal{L}$-subconstructive ordinals, and τ is not $\mathcal{L}$-subconstructive. When we define G_τ we are in case 1, subcase II. The answer to the question is no. Let $x \in C_\Theta$ (= the set of convergent Θ-computations), $|x|_\Theta = \eta_\nu$, $x \in S$. Then $G(f_{xy}) = 1$ for all y such that $|y| = \tau$. (Such a y exists because there are $\mathcal{L}$-computations with length greater than all $\mathcal{L}$-subconstructive ordinals.) Also τ is the least ordinal such that $\exists y[\,|y|^{\mathcal{L}} = \tau$ and $G(f_{xy}) = 1]$.

There are indexes e_1, e_2 such that $G(f_{xy}) = 1$ iff $\{e_1\}^{\mathcal{L}}(x,y)\downarrow$, and if $G(f_{xy}) = 1$ then $|y|^{\mathcal{L}} \leq |\{e_1\}^{\mathcal{L}}(x,y)|$. For all x,y: $G(f_{xy}) \simeq \{e_2\}^{\mathcal{L}}(x,y)$, and if $G(f_{xy}) = 1$ then $|\{e_1\}^{\mathcal{L}}(x,y)| \leq |\{e_2\}^{\mathcal{L}}(x,y)|$.

Let $Q = \{\langle e,a\rangle : \{e\}^{\mathcal{L}}(a)\downarrow$, a is a list of objects from $S\}$. By theorem 8 the ordinal $\kappa^{Q,x}$ is x-reflecting. By the remark following theorem 8 $\kappa^{S,x} < \kappa^{Q,x}$. $\kappa^{S,x} = \kappa^S$ since $x \in S$. By assumption $\tau = \kappa^S$.

Let m be an index such that $\{m\}^{\mathcal{L}}(Q,x)\downarrow$ and $\kappa^S = \kappa^{S,x} < |\{m\}\,(Q,x)|$. (Such an m exists since $\kappa^{S,x} < \kappa^{Q,x}$.) Then $\exists y\,[\,|y|^{\mathcal{L}} < |\{m\}^{\mathcal{L}}(Q,x)|$ and $G(f_{xy}) = 1]$ (let $|y|^{\mathcal{L}} = \tau$). This expression defines a relation $R(Q,x)$, and R is recursively enumerable in $\mathcal{L}$. An index for the expression inside the brackets can be found by the instructions: First see if $|y|^{\mathcal{L}} < |\{m\}^{\mathcal{L}}(Q,x)|$. If false then stop. If true then compute $\{e_2\}^{\mathcal{L}}(x,y)$. For Q and x such that $\{m\}^{\mathcal{L}}(Q,x)\downarrow$ this defines a total relation of y. Hence the quantifier $\exists y$ can be expressed by E. An index e' for R can be found such that $\{e'\}^{\mathcal{L}}(Q,x)\downarrow$, and if $|y|^{\mathcal{L}} < |\{m\}^{\mathcal{L}}(Q,x)|$ and $G(f_{xy}) = 1$ (for instance when $|y|^{\mathcal{L}} = \tau$) then $|\{e_1\}^{\mathcal{L}}(x,y)| < |\{e'\}^{\mathcal{L}}(Q,x)|$.

Now $|\{e'\}^{\mathcal{L}}(Q,x)| < \varkappa^{Q,x}$. Hence $\exists y\,|\{e_1\}^{\mathcal{L}}(x,y)| < \varkappa^{Q,x}$ (let $|y|^{\mathcal{L}} = \tau$). By reflection $\exists y\,|\{e_1\}^{\mathcal{L}}(x,y)| < \varkappa^{X}$. $\varkappa^X \leq \varkappa^S = \tau$, and $|y|^{\mathcal{L}} \leq |\{e_1\}^{\mathcal{L}}(x,y)|$. Hence $\exists y\,[\,|y|^{\mathcal{L}} < \tau$ and $\{e_1\}^{\mathcal{L}}(x,y)\downarrow]$, and $\exists y\,[\,|y|^{\mathcal{L}} < \tau$ and $G(f_{xy}) = 1]$. This is contrary to the fact that τ is the least ordinal such that $\exists y[\,|y|^{\mathcal{L}} = \tau$ and $G(f_{xy}) = 1]$.

$\square$

<u>Proposition 3</u>: $S\text{-en}(\Theta) \subseteq S\text{-en}(\mathcal{L})$; $sc(\Theta,r) \subseteq sc(\mathcal{L},r)$ for all $r \in S$.

Proof: Let $H_\Theta = \{\langle e,a\rangle : |\{e\}_\Theta(a)|_\Theta < \eta_\nu$ for some $\nu < \lambda\}$. Suppose $X \subseteq S$, $X \in S\text{-en}(\Theta)$. Then there is an index e such that $r \in X \iff \{e\}_\Theta(r)\downarrow$. Hence $r \in X \iff \langle e,r\rangle \in H_\Theta$.

Suppose $X \subseteq A$, $X \in sc(\Theta,r)$, $r \in S$. Then there are indexes e_1, e_2 such that $x \in X \iff \{e_1\}_\Theta(x,r)\downarrow$, $x \notin X \iff \{e_2\}_\Theta(x,r)\downarrow$. There is an index e' such that $\lambda x\{e'\}_\Theta(x,r)$ is the characteristic function of X, and for all $x \in X$: $|\{e_1\}_\Theta(x,r)| < |\{e'\}_\Theta(x,r)|$; for all $x \notin X$: $|\{e_2\}_\Theta(x,r)| < |\{e'\}_\Theta(x,r)|$. Let e be an index for the computation $E(\lambda x\{e'\}_\Theta(x,r))$. Then $\{e\}_\Theta(r)\downarrow$, and for all $x \in X$: $|\{e_1\}_\Theta(x,r)| < |\{e\}_\Theta(r)|$, for all $x \notin X$: $|\{e_2\}_\Theta(x,r)| < |\{e\}_\Theta(r)|$. Now $|\{e\}_\Theta(r)|_\Theta = \eta_\nu$ for some $\nu < \lambda$. Hence $x \in X \iff \langle e_1,x,r\rangle \in H_\Theta$, $x \notin X \iff \langle e_2,x,r\rangle \in H_\Theta$.

It is enough to prove that $H_\Theta \in en(\mathcal{L})$, for it follows from the discussion above that if this is true then $S\text{-en}(\Theta) \subseteq S\text{-en}(\mathcal{L})$, and $sc(\Theta,r) \leq sc(\mathcal{L},r)$ for all $r \in S$. By proposition 2 $x \in H_\Theta \iff \exists y \in S\ (y \in C^{\mathcal{L}}$ and $G(f_{xy}) = 1)$. Hence $H_\Theta \in en(\mathcal{L})$.

$\square$

Let $\eta = \sup\{\eta_\nu : \nu < \lambda\}$ $(= \varkappa^S_\Theta)$, and let $\sigma = \sup\{\sigma_\nu : \nu < \lambda\}$.

<u>Proposition 4</u>: a) There are a total Θ-computable function f and

a Θ-computable partial function p such that $\{f(e)\}_\Theta(a) \simeq \{e\}^{\mathcal{L}}(a)$ for all e, a such that $|\{e\}^{\mathcal{L}}(a)| < \sigma$. If $|x|^{\mathcal{L}} < \sigma$ or $|y|^{\mathcal{L}} < \sigma$ then $p(x,y)\downarrow$. If $x \in C^{\mathcal{L}}$, $|x|^{\mathcal{L}} < \sigma$ and $|x|^{\mathcal{L}} \leq |y|^{\mathcal{L}}$, then $p(x,y) \simeq 0$. If $|y|^{\mathcal{L}} < \sigma$ and $|x|^{\mathcal{L}} > |y|^{\mathcal{L}}$ then $p(x,y) \simeq 1$.

b) There are a total Θ-computable function f' and a Θ-computable partial function p' such that $\{f'(e)\}_\Theta(\grave{a},P) \simeq \{e\}^{\mathcal{L}}(a)$ for all e, a. If $|x|^{\mathcal{L}} < \kappa^{\mathcal{L}}$ or $|y|^{\mathcal{L}} < \kappa^{\mathcal{L}}$ then $p'(x,y,P)\downarrow$, $p'(x,yP) \simeq 0$ if $x \in C^{\mathcal{L}}$ and $|x|^{\mathcal{L}} \leq |y|^{\mathcal{L}}$. $p'(x,y,P) \simeq 1$ if $|x|^{\mathcal{L}} > |y|^{\mathcal{L}}$.

Proof: a) Θ-indexes for f and p can be found by the second recursion theorem. We give the main points in the construction of these indexes. Let $\mu < \sigma$. Suppose that $\{f(e)\}_\Theta(a) \simeq \{e\}^{\mathcal{L}}(a)$ for all e, a such that $|\{e\}^{\mathcal{L}}(a)|^{\mathcal{L}} < \mu$, and that $p(x,y)$ is defined and has the right value when $\inf(|x|^{\mathcal{L}}, |y|^{\mathcal{L}}) < \mu$. When $|\{e\}^{\mathcal{L}}(a)|^{\mathcal{L}} = \mu$ we describe $\{f(e)\}_\Theta(a)$ in terms of $\{f(e')_\Theta(a')$ and $p(x',y')$, where $\{e'\}^{\mathcal{L}}(a')$ is an immediate subcomputation of $\{e\}^{\mathcal{L}}(a)$ and $\inf(|x'|^{\mathcal{L}}, |y'|^{\mathcal{L}}) < \mu$. When $\inf(|x|^{\mathcal{L}}, |y|^{\mathcal{L}}) = \mu$ we also describe $p(x,y)$ in terms of $p(x',y')$, $\{f(e')\}_\Theta(a')$, where $\inf(|x'|^{\mathcal{L}}, |y'|^{\mathcal{L}}) < \mu$ and $|\{e'\}^{\mathcal{L}}(a')|^{\mathcal{L}} < \mu$.

Let $|\{e\}^{\mathcal{L}}(a)|^{\mathcal{L}} = \mu$. If $\{e\}^{\mathcal{L}}(a)$ is not an application of G it is obvious how to define $f(e)$. So suppose $\{e\}^{\mathcal{L}}(a) \simeq G(\lambda u\{e'\}^{\mathcal{L}}(u,a))$, where $|\{e'\}^{\mathcal{L}}(u,a)| < \mu$ for all u. By the induction hypothesis $\{f(e')\}_\Theta(u,a) \simeq \{e'\}^{\mathcal{L}}(u,a)$ for all u. To find the value of $G(\lambda u\{e'\}^{\mathcal{L}}(u,a))$ we must see if $\lambda u\{e'\}^{\mathcal{L}}(u,a) = f_{xy}$ for some x, y, and if this is true find the value of $G(f_{xy})$. We do this in five questions. Notice that by the construction of f_{xy} $\lambda u\{e'\}^{\mathcal{L}}(u,a) \neq f_{xy}$ if $|y|^{\mathcal{L}} \geq \mu$.

First question: Are there x, y such that $|y|^{\mathcal{L}} < \mu$ and

$\lambda u\{e'\}^{\mathcal{L}}(u,a) = f_{xy}$?

No : $G(\lambda u\{e'\}^{\mathcal{L}}(u,a)) \simeq 0$,

Yes: go on to the second question.

Second question: Let $\tau < \mu$ be the ordinal such that for some x and y: $\tau = |y|^{\mathcal{L}}$ and $\lambda u\{e'\}^{\mathcal{L}}(u,a) = f_{xy}$.

Is there an ordinal $\nu < \lambda$ such that $\sigma_\rho < \tau$ when $\rho < \nu$, and $\sigma_\nu \geq \tau$?

No : $G(\lambda u\{e'\}^{\mathcal{L}}(u,a)) \simeq 0$,

Yes: go on to the third question.

Third question: Let ν, τ be as above. Is there an x such that $|x|_\Theta \leq \eta_\nu$ and $\lambda u\{e'\}^{\mathcal{L}}(u,a) = f_{xy}$, where $|y|^{\mathcal{L}} = \tau$?

No : $G(\lambda u\{e'\}^{\mathcal{L}}(u,a)) \simeq 0$,

Yes: go on to the fourth question.

Fourth question: Is τ $\mathcal{L}$-subconstructive?

Yes: $G(\lambda u\{e'\}^{\mathcal{L}}(u,a)) \simeq 1$,

No : go on to the fifth question.

Fifth question: Let $\epsilon = \eta_\nu - \sup\{\eta_\rho : \rho < \nu\}$. Is there an ordinal π such that $\tau < \pi \leq \tau + \epsilon$ and π is $\mathcal{L}^0$-subconstructive?

Yes: $G(\lambda u\{e'\}^{\mathcal{L}}(u,a)) \simeq 0$,

No : $G(\lambda u\{e'\}^{\mathcal{L}}(u,a)) \simeq 1$.

Next we examine the first two questions to find Θ-indexes for them. The examination of the last three questions will be omitted.

First question. $|y|^{\mathcal{L}} < \mu \iff \exists u(|y|^{\mathcal{L}} \leq |\{e'\}^{\mathcal{L}}(u,a)|)$. By the induction hypothesis $|y|^{\mathcal{L}} < \mu \iff \exists u\ p(y,\langle e',u,a\rangle) \simeq 0$. $\lambda u\ p(y,\langle e\ ,u,a\rangle)$ is total. Hence the quantifier $\exists u$ can be expressed by E, and the relation " $|y|^{\mathcal{L}} < \mu$ " is Θ-computable, uniformly in e, a. Let e_1 be a Θ-index for the characteristic function of this relation. To describe f_{xy} we need all information about the $\mathcal{L}$-computations of length $< |y|^{\mathcal{L}}$. By the induction hypothes this information can be obtained from $\lambda e a \{f(e)\}_{\Theta}(a)$ and p when $|y|^{\mathcal{L}} < \mu$. Hence there is an index e_2 such that $f_{xy} = \lambda u\{e_2\}_{\Theta}(u,x,y,f(e),a)$ when $|y|^{\mathcal{L}} < \mu$. In the first question we ask if the following statement is true: $\exists x\, \exists y\, (|y|^{\mathcal{L}} < \mu$ and $f_{xy} = \lambda u\{f(e')\}_{\Theta}(u,a))$. A Θ-index for the relation inside the brackets can be found from e_1 and e_2. The quantifers $\exists x\, \exists y$ can be expressed by E.

Second question: Since $\mu < \sigma$ the answer to this question is yes. So to compute $G(\lambda u\{e'\}^{\mathcal{L}}(u,a))$ to on to question 3.

Description of $p(x,y)$. Suppose $\inf(|x|^{\mathcal{L}}, |y|^{\mathcal{L}}) = \mu$. The definition of $p(x,y)$ is by cases. The form of x and y determines which case we are in. Only one case will be studied here: When both x and y correspond to substitutions, i.e. $x = \langle\langle 10,n,g,h\rangle,a\rangle$, $y = \langle\langle 10,n',g',h'\rangle,a'\rangle$. If x is convergent then the immediate subcomputations of x are $\langle g,a\rangle$ and $\langle h,\{g\}^{\mathcal{L}}(a),a\rangle$. If y is convergent then the immediate subcomputations of y are $\langle g',a'\rangle$ and $\langle h',\{g'\}^{\mathcal{L}}(a'),a'\rangle$.

$$
\begin{aligned}
p(x,y) \simeq 0 \quad & \text{if} \quad |\{g\}^{\mathcal{L}}(a)|,\ |\{h\}^{\mathcal{L}}(\{g\}^{\mathcal{L}}(a),a)| \le |\{g'\}^{\mathcal{L}}(a')| \\
& \text{or} \quad |\{g\}^{\mathcal{L}}(a)|,\ |\{h\}^{\mathcal{L}}(\{g\}^{\mathcal{L}}(a),a)| \le |\{h'\}^{\mathcal{L}}(\{g'\}^{\mathcal{L}}(a'),a')|, \\
\simeq 1 \quad & \text{if} \quad |\{g\}^{\mathcal{L}}(a)| > |\{g'\}^{\mathcal{L}}(a')|,\ |\{h'\}^{\mathcal{L}}(\{g'\}^{\mathcal{L}}(a'),a')| \\
& \text{or} \quad |\{h\}^{\mathcal{L}}(\{g\}^{\mathcal{L}}(a),a)| > |\{g'\}^{\mathcal{L}}(a')|,\ |\{h'\}^{\mathcal{L}}(\{g'\}^{\mathcal{L}}(a'),a')|.
\end{aligned}
$$

"$|\{g\}^{\mathcal{L}}(a)|,\ |\{h\}^{\mathcal{L}}(\{g\}^{\mathcal{L}}(a),a)| \le |\{g'\}^{\mathcal{L}}(a')|$" can be expressed as: "$p(\langle g,a\rangle,\langle g',a'\rangle) \simeq 0$ and $p(\langle h,\{f(g)\}_{\Theta}(a),a\rangle,\langle g',a\rangle) \simeq 0$". If $|x|^{\mathcal{L}} \le |y|^{\mathcal{L}}$ then by the induction hypothesis $p(\langle g,a\rangle,\langle g',a'\rangle)$ is defined, $\{f(g)\}_{\Theta}(a) \simeq \{g\}^{\mathcal{L}}(a)$, $\{g\}^{\mathcal{L}}(a)\downarrow$, and $p(\langle h,\{f(g)\}_{\Theta}(a),a\rangle,\langle g',a'\rangle)$ is defined. The other parts of the definition of p can be replaced by similar expressions. This describes $p(x,y)$ in terms of $p(x',y')$, where $\inf(x',y') < \mu$, and $\{f(e')\}_{\Theta}(a')$ where $|\{e'\}^{\mathcal{L}}(a')| < \mu$.

This proves a). Note that $\{f(e)\}_{\Theta}(a) \simeq \{e\}^{\mathcal{L}}(a)$ may not be true if $|\{e\}^{\mathcal{L}}(a)| \geq \sigma$, because in this case the answer to question 2 can be no. In the construction of f we assumed that the answer was yes. Because of this $p(x,y)$ may not have the right value when $\inf(|x|^{\mathcal{L}},|y|^{\mathcal{L}}) > \sigma$. For f occurs in the description of $p(x,y)$. In the case given above f occurs in the expression $\{f(g)\}_{\Theta}(a)$.

Proof of b) . Θ-indexes for f' and p' can be found by the second recursion theorem. The construction is similar to the construction in a) . Let $\mu < \kappa^{\mathcal{L}}$. Suppose $\{f'(e)\}_{\Theta}(a,P) \simeq \{e\}^{\mathcal{L}}(a)$ for all e, a such that $|\{e\}^{\mathcal{L}}(a)| < \mu$, and $p'(x,y,P)$ is defined and has the right value when $\inf(|x|^{\mathcal{L}},|y|^{\mathcal{L}}) < \mu$. As in a) we describe $\{f'(e)\}_{\Theta}(a,P)$ when $|\{e\}^{\mathcal{L}}(a)| = \mu$, and $p'(x,y,P)$ when $\inf(|x|^{\mathcal{L}},|y|^{\mathcal{L}}) = \mu$. $p'(x,y,P)$ is defined from f' in the same way as p was defined from f . So it is enough to regard $\{f'(e)\}_{\Theta}(a,P)$ where $|\{e\}^{\mathcal{L}}(a)| = \mu$. As in a) we only regard the case $\{e\}^{\mathcal{L}}(a) \simeq G(\lambda u\{e'\}^{\mathcal{L}}(u,a))$. The description of $\{f'(e)\}_{\Theta}(a,P)$ is the same as the description of $\{f(e)\}_{\Theta}(a)$ in a) up to the second question. Here the descriptions differ, as we cannot assume that the answer is yes. Below follows the examination of the second question.

Let $Q = \{r : r \in S$ and r is a convergent $\mathcal{L}$-computation$\}$. If $r \in Q$ let $|r|_Q$ be the ordinal ν such that $|r|^{\mathcal{L}} = \sigma_\nu$. Hence $|r|_Q$ is the order type of the $\mathcal{L}$-subconstructive ordinals $< |r|^{\mathcal{L}}$. If $r \notin Q$ let $|r|_Q$ be the order type of the $\mathcal{L}$-subconstructive ordinals. The set P was defined in the beginning of the proof of the theorem. If $r \in P$ let $|r|_P$ be the ordinal ν such that $|r|_{\Theta} = \eta_\nu$. If $r \notin P$ let $|r|_P = \lambda$. From the result in a) one can deduce that there is an index e_1 such that if $|s|_Q < \lambda$, $|s|_Q \leq |s'|_Q$ then $\{e_1\}_{\Theta}(s,s') \simeq 0$, if $|s'|_Q < \lambda$, $|s|_Q > |s'|_Q$ then $\{e_1\}_{\Theta}(s,s') \simeq 1$. From e_1 one can construct an index e_2 such that if $r \in P$, $|r|_P \leq |s|_Q$ then $\{e_2\}_{\Theta}(r,s) \simeq 0$, if $|r|_P > |s|_Q$ then $\{e_2\}_{\Theta}(r,s) \simeq 1$. (e_1 and e_2 can be found by applications of the second recursion theorem.)

The following is an equivalent reformulation of the second question: Is there $r \in P$ such that $|r|_P$ is the order type of the $\mathcal{L}$-subconstructive ordinals $< |y|^{\mathcal{L}}$? A Θ-index for " $r \in P$ and $|r|_P$ is the order type of the $\mathcal{L}$-subconstructive ordinals $< |y|^{\mathcal{L}}$ " can be found from the following instructions: First see if $r \in P$. If not give output 1. If $r \in P$ let ν be the order type of the $\mathcal{L}$-subconstructive ordinals $< |y|^{\mathcal{L}}$. It remains to decide whether or not $|r|_P = \nu$. $|r|_P \leq \nu \iff \forall r'[|r'|_P < |r|_P \implies \exists s(|s|^{\mathcal{L}} < |y|^{\mathcal{L}}$ and $|r'|_P \leq |s|_Q]$. By the induction hypothesis $\lambda s(1 - p'(y,s,P))$ is the characteristic function of " $|s|^{\mathcal{L}} < |y|^{\mathcal{L}}$ " as a relation of s. $\lambda s\{e_2\}_{\Theta}(r',s)$ is the characteristic function of " $|r'|_P \leq |s|_Q$ " as a relation of s when $r' \in P$. The quantifiers $\forall r'$, $\exists s$ can be expressed by the functional E. This proves that there is an index e_3, such that $\lambda r\{e_3\}_{\Theta}(r,y,P)$ is the characteristic function of $|r|_P \leq \nu$. In a similar way one can decide whether or not $|r|_P \geq \nu$. Hence there is an index e_4 such that $\{e_4\}_{\Theta}(y,P)$ gives the answer to the second question. $\square$

<u>Remark 1</u>: In the examination of question 2 we ask whether or not $r \in P$. Hence P is used negatively. This is the only place where

it is necessary to regard P as an argument in the computation. In all other parts of the construction P is used positively. Hence expressions like $r \in P$ can be replaced by $\{e\}_{\Theta}(r)\downarrow$, where e is a Θ-index for P.

Remark 2: Let $\lambda' \leq \lambda$. Let $P' = P_{\lambda'} = \{r \in P : |r|_P < \lambda'\}$. Let $\eta' = \sup\{\eta_\rho + 1 : \rho < \lambda'\}$. If we replace P by P' in the construction of G we will obtain another functional G'. Let $\mathcal{L}' = G', E$. The constructions of G and G' are equal up to stage $\sigma' = \sup\{\sigma_\rho + 1 : \rho < \lambda'\}$, i.e. $G'_\mu = G_\mu$ for $\mu < \sigma'$. The functions f', p' from part b) of proposition 4 have the properties: If $\{e\}^{\mathcal{L}'}(a)\downarrow$ then $\{f'(e)\}_{\Theta}(a,P') \simeq \{e\}^{\mathcal{L}'}(a)$. If $x \in C^{\mathcal{L}'}$ and $|x|^{\mathcal{L}'} \leq |y|^{\mathcal{L}'}$ then $p'(x,y,P') \simeq 0$, if $|x|^{\mathcal{L}'} > |y|^{\mathcal{L}'}$ then $p'(x,y,P') \simeq 1$.

Proposition 5: The order type of the $\mathcal{L}$-subconstructive ordinals is λ.

Proof: By proposition 2 the order type $\geq \lambda$. Suppose that the order type $> \lambda$. Then there is $s_0 \in Q$ such that $|s_0|^{\mathcal{L}} = \sigma_\lambda$. By reflection we will prove that $|s_0|^{\mathcal{L}} < \sigma_\lambda$, and hence obtain a contradiction. Below follow some technical preliminaries in order to reflect.

There are indexes e_1, e_2, e_3, e_4 such that $\{e_1\}_{\Theta}(s_0,P')\downarrow$ iff s_0 is a convergent $\mathcal{L}'$-computation and λ' is the order type of the $\mathcal{L}'$-subconstructive ordinals $< |s_0|^{\mathcal{L}'}$. $\{e_2\}_{\Theta}(s_0,P')\downarrow$ iff s_0 is a convergent $\mathcal{L}'$-computation, in which case $|s_0|^{\mathcal{L}'} < |\{e_2\}_{\Theta}(s_0,P')|_{\Theta}$, $\{e_3\}_{\Theta}(P')\downarrow$ and $\eta' \leq |\{e_3\}_{\Theta}(P')|_{\Theta}$. $\{e_4\}_{\Theta}(s_0,P')\downarrow$ iff s_0 is a convergent $\mathcal{L}'$-computation, in which case $\eta' + |s_0|^{\mathcal{L}'} < |\{e_4\}_{\Theta}(s_0,P')|_{\Theta}$. ($P', \lambda', \eta', \mathcal{L}'$ are as in remark

2 above.)

e_1 can be constructed in the following way: The statement " s_o is a convergent $\mathscr{L}'$-computation " is Θ-semicomputable in P'. An index for this statement can be found from an index for p' in part b) of proposition 4. The statement " λ' is the order type of the $\mathscr{L}'$-subconstructive ordinals $< |s_o|^{\mathscr{L}'}$ " can be reformulated by: $\forall r \in P' \exists s(|r|_P = |s|_Q$ and $|s|^{\mathscr{L}'} < |s_o|^{\mathscr{L}'})$ and $\forall s(|s|^{\mathscr{L}'} < |s_o|^{\mathscr{L}'} \Rightarrow \exists r \in P'(|r|_P = |s|_Q))$. By an application of the second recursion theorem and of the result in part a) of proposition 4 it can be proved that the relation " $r \in P$ and $s \in Q$ and $|r|_P = |s|_Q$ " is Θ-semicomputable. This relation can be used to express " $|r|_P = |s|_Q$ " in the statement above. The other parts of the statement can be expressed by the function p' in part b) of proposition 4. The quantifiers can be expressed by E.

e_2 can be constructed in the following way: Let $s_o = \langle e,a \rangle$. Then $s_o \in C^{\mathscr{L}'}$ iff $\{e\}^{\mathscr{L}'}(a)\downarrow$ iff $\{f'(e)\}_\Theta(a,P')\downarrow$. The function f' in proposition 4 b) can be chosen such that the following is true: If $\{e\}^{\mathscr{L}'}(a)\downarrow$ then $|\{e\}^{\mathscr{L}'}(a)|^{\mathscr{L}'} < |\{f'(e)\}_\Theta(a,P')|_\Theta$. Let $\{e_2\}_\Theta(s_o,P') \simeq \{f'(e)\}_\Theta(a,P')$.

Let e_3 be an index for the following instruction: For all $r \in P'$ compute the computation r. Then $\{e_3\}_\Theta(P')\downarrow$, and $|r|_\Theta < |\{e_3\}_\Theta(P')|_\Theta$ for all $r \in P'$. Hence $\eta' \leq |\{e_3\}(P')|_\Theta$.

From e_2 and e_3 one can construct an index e_4 such that $\{e_4\}_\Theta(s_o,P')\downarrow$ iff $\{e_2\}_\Theta(s_o,P')\downarrow$ and $\{e_3\}_\Theta(P')\downarrow$, in which case $|\{e_3\}_\Theta(P')|_\Theta + |\{e_2\}_\Theta(s_o,P')| < |\{e_4\}_\Theta(s_o,P')|_\Theta$. Hence $\{e_4\}_\Theta(s_o,P')\downarrow$ iff s_o is a convergent $\mathscr{L}'$-computation, in which case $\eta' + |s_o|^{\mathscr{L}'} < |\{e_4\}_\Theta(s_o,P')|_\Theta$.

Now we come to the reflecting statement. It is a conjunction of three parts a), b) and c).

a) $\exists\lambda' \leq \lambda \quad (x = P_{\lambda'})$.

b) s_o is a convergent $\mathcal{L}'$-computation, and λ' is the order type of the $\mathcal{L}'$-subconstructive ordinals $< |s_o|^{\mathcal{L}'}$.

c) If $r \notin x$ then $\eta' + |s_o|^{\mathcal{L}'} < |r|_\Theta$.

There is an index e_5 such that if $\{e_5\}_\Theta(s_o,x)\downarrow$ then a), b) and c) are satisfied for x. e_5 can be constructed as follows: a) can be expressed by " $x \subseteq S$, and each element in x is a convergent Θ-computation, and for all r,r': If $r \in x$ and $|r'|_\Theta \leq |r|_\Theta$ then $r' \in x$ ". A Θ-index for this statement can be found. If $x = P_{\lambda'} = P'$ then b) is satisfied iff $\{e_1\}_\Theta(s_o,x)\downarrow$. The following statement implies c) when $x = P'$: "for all $r \notin x$ $|\{e_4\}_\Theta(s_o,x)| < |r|_\Theta$ ". A Θ-index for this statement can be found. Let e_5 be a Θ-index for the conjunction of these statements.

Let B be defined by: $x \in B \iff \{e_5\}_\Theta(s_o,x)\downarrow$. Then B is Θ-semicomputable in s_o, and by a) in the reflecting statement each element in B is a subset of S (in fact $x \in B \Rightarrow x = P_\nu$ for a $\nu \leq \lambda$). $P \in B$, for a) is trivially satisfied for $x = P$. By assumption b) is satisfied for $x = P$. To see that c) is satisfied for $x = P$ let $r \notin P$. Then r is a divergent Θ-computation, and $|r|_\Theta = \varkappa_\Theta$. Now $\{e_4\}_\Theta(s_o,x)\downarrow$, hence $|\{e_4\}_\Theta(s_o,x)|_\Theta \leq |r|_\Theta$, and $\{e_5\}_\Theta(s_o,x)\downarrow$. P is Θ-semicomputable. By the corollary of theorem 8 there is an element x_o in B which is Θ-computable in s_o. Since P is not Θ-computable in s_o it follows that $x_o = P_{\lambda'}$ where $\lambda' < \lambda$. Let $P' = P_{\lambda'}$.

Since the reflecting statement is true for P': s_o is a convergent $\mathcal{L}'$-computation, and λ' is the order type of the $\mathcal{L}'$-subconstructive ordinals $< |s_o|^{\mathcal{L}'}$, and if $r \notin P'$ then $\eta' + |s_o|^{\mathcal{L}'} < |r|_\Theta$.

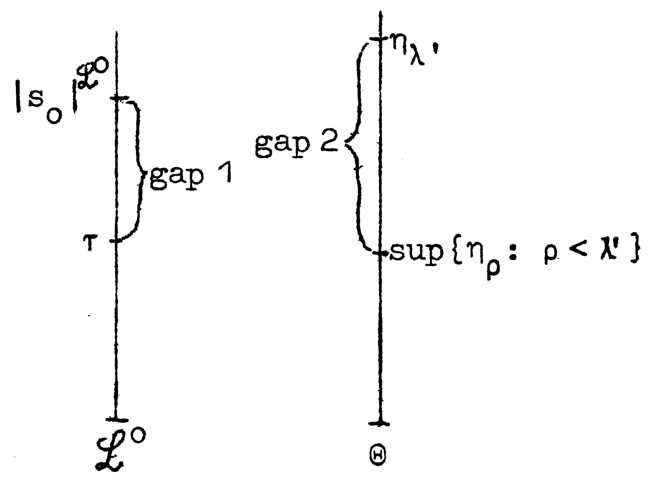
To prove that $|s_0|^{\mathcal{L}} < \lambda$ we go back to the constructions of G and G'. As mentioned in remark 2 $G'_\mu = G_\mu$ when $\mu < \sigma'$ where $\sigma' = \sup\{\sigma_\rho + 1 : \rho < \lambda'\}$. Let $\tau = \sigma'$. The $\mathcal{L}$-subconstructive and the $\mathcal{L}'$-subconstructive ordinals $< \tau$ are the same, and the order type of these ordinals is λ'. τ is $\mathcal{L}$-subconstructive iff τ is $\mathcal{L}'$-subconstructive. By b) in the reflecting statement $|s_0|^{\mathcal{L}'}$ is the λ'-th $\mathcal{L}'$-subconstructive ordinal.

<u>Claim</u>: $G'_\mu = G_\mu$ for all $\mu < |s_0|^{\mathcal{L}'}$.

Proof: If $\tau = |s_0|^{\mathcal{L}'}$ (which is the case of τ is $\mathcal{L}$-subconstructive) then the claim is true by the discussion above. Suppose $\tau < |s_0|^{\mathcal{L}'}$. This is the case if τ is not $\mathcal{L}$-subconstructive. In the construction of G'_τ we are in case 2 for the first time. Hence

$$G'(f) = \begin{cases} (\bigcup_{\mu<\tau} G_\mu)(f) & \text{if } \exists \mu < \tau \quad f \in \operatorname{dom} G_\mu \\ 0 & \text{otherwise .} \end{cases}$$

In the construction of G_τ we are in case 1 since the order type of the $\mathcal{L}$-subconstructive ordinals less that τ is λ', and $\lambda' < \lambda$. We are in subcase II, because τ is not $\mathcal{L}$-subconstructive by the assumption $\tau < |s_0|^{\mathcal{L}'}$. The functional G^0 mentioned at this stage of the construction is the same as G'. Hence s_0 is a convergent $\mathcal{L}^0$-computation, $|s_0|^{\mathcal{L}'} = |s_0|^{\mathcal{L}^0}$, and $|s_0|^{\mathcal{L}^0}$ is the first ordinal $> \tau$ which is $\mathcal{L}^0$-subconstructive. Let $\varepsilon = \eta_{\lambda'} - \sup\{\eta_\rho : \rho < \lambda'\}$. By c) in the reflecting statement $\eta' + |s_0|^{\mathcal{L}^0} < |r|_\Theta$ if $r \notin P'$, i.e. if $|r|_\Theta \geq \eta_{\lambda'}$. Hence $\eta' + |s_0|^{\mathcal{L}^0} < \eta_{\lambda'}$. By definition $\eta' = \sup\{\eta_\rho + 1 : \rho < \lambda'\}$. Hence $\sup\{\eta_\rho : \rho < \lambda'\} + |s_0|^{\mathcal{L}^0} < \eta_{\lambda'}$.



Hence $\sup\{\eta_\rho : \rho < \lambda'\} + (|s_o|^{\mathcal{L}^o} - \tau) < \eta_{\lambda'}$,

Hence gap 1 < gap 2, where gap 1 $= |s_o|^{\mathcal{L}^o} - \tau$, and gap 2 $= \epsilon$. So the following inequality is true: $\tau < |s_o|^{\mathcal{L}^o} < \tau + \epsilon$. Hence there is an ordinal π such that $\tau < \pi \leq \tau + \epsilon$, and π is $\mathcal{L}^o$-subconstructive. So the answer to the question in the construction of G_τ is yes. Hence $G_\tau = G'_\tau$, and in fact $G_\mu = G'_\mu$ for all μ less that the next $\mathcal{L}^o$-subconstructive ordinal, which is $|s_o|^{\mathcal{L}'}$. This proves the claim.

By the claim the $\mathcal{L}'$-computations of length less that $|s_o|^{\mathcal{L}'}$ are identical to the $\mathcal{L}$-computations of length less that $|s_o|^{\mathcal{L}'}$. If $s_o = \langle e,a \rangle$ then $\{e\}^{\mathcal{L}'}(a)\downarrow$. All immediate subcomputations $\{e'\}^{\mathcal{L}'}(a')$ of $\{e\}^{\mathcal{L}'}(a)$ have length less that $|s_o|^{\mathcal{L}'}$, hence $\{e'\}^{\mathcal{L}'}(a') \simeq \{e'\}^{\mathcal{L}}(a')$, and $|\{e'\}^{\mathcal{L}}(a')|^{\mathcal{L}} < |s_o|^{\mathcal{L}'}$. Hence $\{e\}^{\mathcal{L}}(a)\downarrow$, and $|\{e\}^{\mathcal{L}}(a)|^{\mathcal{L}} = |s_o|^{\mathcal{L}'}$, i.e. $|s_o|^{\mathcal{L}} = |s_o|^{\mathcal{L}'} = \sigma_{\lambda'}$, a contradiction. This proves proposition 5. $\square$

<u>Proposition 6</u>: $S - \mathrm{en}(\mathcal{L}) \subseteq S - \mathrm{en}(\Theta)$; and for all $r \in S$: $\mathrm{sc}(\mathcal{L},r) \subseteq \mathrm{sc}(\Theta,r)$.

Proof: Let $H^{\mathcal{L}} = \{\langle e,a \rangle : \{e\}^{\mathcal{L}}(a)\downarrow$ and $|\{e\}^{\mathcal{L}}(a)|^{\mathcal{L}} < \sigma\}$. By proposition 5 σ is the supremum of the $\mathcal{L}$-subconstructive ordinals. If $X \subseteq S$, $X \in S - \mathrm{en}(\mathcal{L})$ then there is an index e such that for all r: $r \in X \iff \langle e,r \rangle \in H^{\mathcal{L}}$.

Suppose $X \subset A$, $X \in \mathrm{sc}(\mathcal{L},r)$, $r \in S$. By the method used in the proof of proposition 3 one can prove that there are indexes

e_1, e_2 such that for all x: $x \in X \iff \langle e_1,x,r\rangle \in H^{\mathcal{L}}$, $x \notin X \iff \langle e_2,x,r\rangle \in H^{\mathcal{L}}$.

So it is enough to prove that $H^{\mathcal{L}} \in en(\Theta)$. $x \in H^{\mathcal{L}} \iff \exists r$ ($r \in P$ and $|r|_P$ is the order type of the $\mathcal{L}$-subconstructive ordinals $< |x|^{\mathcal{L}}$). By an application of the second recursion theorem and of the result in part a) of proposition 4 one can prove that the relation " $r \in P$ and $|r|_P$ is the order type of the $\mathcal{L}$-subconstructive ordinals $< |x|^{\mathcal{L}}$ " is Θ-semicomputable. Hence $H^{\mathcal{L}} \in en(\Theta)$. $\square$

Propositions 3 and 6 prove theorem 9. $\square$

In the following pages computation theories on α will be denoted by Θ, Ψ instead of $(\Theta, ||_\Theta)$, $(\Psi, ||_\Psi)$. Let En(Θ), Sc(Θ), Sc(Θ,a) be defined by:

En(Θ) = {φ : φ is a partial Θ-computable function},
Sc(Θ) = {f : f is a total Θ-computable function},
Sc(Θ,a) = {f : f is total Θ-computable in a}.

Let Ψ, Θ be normal computation theories on α. Let $\sim$, $\leq$, $<_1$, $<_2$ be defined by:

$\Psi \sim \Theta \iff En(\Psi) = En(\Theta)$,
$\Psi \leq \Theta \iff En(\Psi) \subseteq En(\Theta)$,
$\Psi <_1 \Theta \iff En(\Psi) \subseteq En(\Theta)$ and $\exists x\, (\kappa_\Psi^x < \kappa_\Theta^x)$,
$\Psi <_2 \Theta \iff En(\Psi) \subsetneq En(\Theta)$.

Lemma 26: Let Ψ and Θ be normal.

a) $\Psi \leq \Theta \Rightarrow \forall a\, (Sc(\Psi,a) \subseteq Sc(\Theta,a))$ and $\forall a (\kappa_\Psi^a \leq \kappa_\Theta^a)$.

b) $\Psi \leq \Psi'$, $\Psi' <_1 \Theta'$, $\Theta' \leq \Theta \Longrightarrow \Psi <_1 \Theta$,

c) $\Psi <_1 \Theta \Longrightarrow \Psi <_2 \Theta$.

Proof: a) Suppose $\Psi \leq \Theta$. Let $f \in Sc(\Psi,a)$. There is an index e such that $f = \lambda x\{e\}_\Psi(x,a)$. Let $\varphi = \lambda x a\{e\}_\Psi(x,a)$. $\varphi \in En(\Psi)$. By assumption $\varphi \in En(\Theta)$. Hence there is an index e' such that $\varphi = \lambda x a\{e'\}_\Theta(x,a)$. Hence $f = \lambda x\{e'\}_\Theta(x,a)$, i.e. $f \in Sc(\Theta,a)$.

$\varkappa_\Psi^a = \sup\{\tau : \tau$ is the length of a prewellordering with domain A which is Ψ-computable in $a\}$. If X is a prewellordering which is Ψ-computable in a then by the above X is Θ-computable in a. Hence $\varkappa_\Psi^a \leq \varkappa_\Theta^a$.

b) Suppose $\Psi \leq \Psi'$, $\Psi' <_1 \Theta'$, $\Theta' \leq \Theta$. Then $En(\Psi) \subseteq En(\Theta)$. Slince $\Psi' <_1 \Theta'$ there is an x such that $\varkappa_{\Psi'}^x < \varkappa_{\Theta'}^x$. By a) $\varkappa_\Psi^x \leq \varkappa_{\Psi'}^x$ and $\varkappa_{\Theta'}^x \leq \varkappa_\Theta^x$. Hence $\varkappa_\Psi^x < \varkappa_\Theta^x$. Hence $\Psi <_1 \Theta$.

c) Suppose $\Psi <_1 \Theta$. Then $En(\Psi) \subseteq En(\Theta)$. If $En(\Psi) = En(\Theta)$ then by a) $\varkappa_\Psi^x = \varkappa_\Theta^x$ for all x, a contradiction. Hence $En(\Psi) \subsetneq En(\Theta)$. □

Let $\mathcal{L}$ be a list, and let Θ be a computation theory. $\mathcal{L}$ is <u>Θ-computable</u> if each function and relation in $\mathcal{L}$ is Θ-computable, and each functional in $\mathcal{L}$ is weakly Θ-computable. Let $\mathcal{P}(\mathcal{L})$ denote the computation theory obtained from $\mathcal{L}$.

<u>Lemma 27</u>: a) If $\mathcal{L}$ is Θ-computable then $\mathcal{P}(\mathcal{L}) \leq \Theta$.
b) If Θ is normal and $\mathcal{P}(\mathcal{L}) \sim \Theta$ then $\mathcal{L}$ is Θ-computable.

Proof: a) Suppose $\mathcal{L}$ is Θ-computable. One can prove that there is a Θ-computable total function f such that for all e, a:

$\{e\}^{\mathcal{L}}(a) \simeq \{f(e)\}_{\Theta}(a)$. Hence $\mathcal{P}(\mathcal{L}) \leq \Theta$.

b) Suppose Θ is normal and $\mathcal{P}(\mathcal{L}) \sim \Theta$. Then the functions and the relations in the list $\mathcal{L}$ are Θ-computable. Let F be a functional in the list $\mathcal{L}$. We must prove that F is weakly Θ-computable. We give instructions how to compute $F(\lambda x\{e\}_{\Theta}(x,a))$. Let $n \geq 0$, and let a range over all lists of length n. Let $\varphi_{\Theta}(e,x,a) \simeq \{e\}_{\Theta}(x,a)$. Then $\varphi_{\Theta} \in En(\Theta)$, hence $\varphi_{\Theta} \in En(\mathcal{L})$. Let $f(n)$ be an $\mathcal{L}$-index for φ_{Θ}. Then $\lambda x\{e\}_{\Theta}(x,a) = \lambda x\{f(n)\}^{\mathcal{L}}(e,x,a)$. $F(\lambda x\{e\}_{\Theta}(x,a)) \simeq F(\lambda x\{f(n)\}^{\mathcal{L}}(e,x,a)) \simeq \{g(n)\}^{\mathcal{L}}(e,a)$, where $g(n)$ can be constructed from $f(n)$ in a primitive recursive way. Let $\varphi_{\mathcal{L}}(e,a) \simeq \{e\}^{\mathcal{L}}(a)$. Then $\varphi_{\mathcal{L}} \in En(\mathcal{L})$, hence $\varphi_{\mathcal{L}} \in En(\Theta)$. So $F(\lambda x\{e\}_{\Theta}(x,a)) \simeq \varphi_{\mathcal{L}}(g(n),e,a)$. There is a Θ-computable total function h such that $F(\lambda x\{e\}_{\Theta}(x,a)) \simeq \{h(n)\}_{\Theta}(e,a)$. To conclude that F is weakly Θ-computable the following should be true: $|\{h(n)\}_{\Theta}(e,a)|_{\Theta} > |\{e\}_{\Theta}(x,a)|_{\Theta}$ for all x. This can be obtained as follows: Let $E(\lambda x\{e\}_{\Theta}(x,a))$ be a subcomputation of $\{h(n)\}_{\Theta}(e,a)$. Since E is weakly Θ- computable we get the right ordinal inequalities. $\square$

<u>Lemma 28</u>: Suppose Θ is normal, $\mathcal{L}$ is Θ-computable and $\forall x(\varkappa_{\mathcal{L}}^{x} = \varkappa_{\Theta}^{x})$. Then there is a normal list $\mathcal{L}'$ such that $\mathcal{P}(\mathcal{L}') \sim \Theta$.

Proof: There is a Θ-index e_1 such that $\{e_1\}_{\Theta}(e,y)\downarrow$ iff $\lambda x\{e\}_{\Theta}(x,y)$ is total, in which case $|\{e\}_{\Theta}(x,y)| < |\{e_1\}_{\Theta}(e,y)|$ for all x. (e_1 can for instance be an index for the computation $E(\lambda x\{e\}_{\Theta}(x,y))$.)

Let f be a variable for total unary functions $A \to S$. Let $Ord(f)$ be the least ordinal τ such that for some e,y: $f = \lambda x\{e\}_{\Theta}(x,y)$, and $|\{e_1\}_{\Theta}(e,y)| = \tau$, if such an ordinal exists.

$\mathrm{Ord}(f)$ is undefined otherwise.

Let G be defined by: If $\mathrm{Ord}(f)$ is defined then

$$G(\langle f,e,a\rangle) = \begin{cases} \{e\}_\Theta(a)+1 & \text{if } |\{e\}_\Theta(a)|_\Theta \le \mathrm{Ord}(f) \\ 0 & \text{otherwise} \end{cases}$$

where $\langle f,e,a\rangle = \lambda x\langle f(x),e,a\rangle$. (We make the following convention: If $r \notin N$ then $r+1 = r$.) $G(g) = 0$ if g is not of the form $\langle f,e,a\rangle$, or if $g = \langle f,e,a\rangle$ and $\mathrm{Ord}(f)$ is undefined.

G is weakly Θ-computable. For let $\varphi = \lambda x\{e'\}_\Theta(x,a')$. To compute $G(\varphi)$ first check if φ is total. (This can for instance be done by computing $E(\varphi)$.) If φ is total check if $\varphi = \langle f,e,a\rangle$ for some f,e,a. If not let $G(\varphi) = 0$. If $\varphi = \langle f,e,a\rangle$ then $\mathrm{Ord}(f)$ is defined since φ is Θ-computable. It remains to see whether or not $|\{e\}_\Theta(a)|_\Theta \le \mathrm{Ord}(f)$. Let $f = \lambda x\{e_2\}_\Theta(x,\langle a'\rangle)$.
$|\{e\}_\Theta(a)|_\Theta \le \mathrm{Ord}(f) \iff \forall h \in N\ \forall y$
$([\forall x(\{h\}_\Theta(x,y) \simeq \{e_2\}_\Theta(x,\langle a'\rangle))$ and $|\{e_1\}_\Theta(h,y)| \le |\{e_1\}_\Theta(e_2,\langle a'\rangle)|]$
$\Rightarrow |\{e\}_\Theta(a)| \le |\{e_1\}_\Theta(h,y)|)$. This expression says:
$\forall h,y\,(f = \lambda x\{h\}_\Theta(x,y) \Rightarrow |\{e\}_\Theta(a)| \le |\{e_1\}_\Theta(h,y)|)$. As a relation of e,a this is Θ-computable in a'. Hence G is weakly Θ-computable. Let $\mathcal{L}' = \mathcal{L},G,E,=_S$, where $=_S$ is the equality relation on S. Then $\mathcal{L}'$ is normal, and $\mathrm{En}(\mathcal{L}') \subseteq \mathrm{En}(\Theta)$ because $\mathcal{L}'$ is Θ-computable. To prove the opposite inclusion we need a proposition.

For all $y \in A$ let $\mu_y = \sup\{\mathrm{Ord}(f) : f$ is recursive in $\mathcal{L}',y\}$.

<u>Proposition 1</u>: $\mu_y = \kappa_\Theta^y$ for all y.

Proof: If f is recursive in $\mathcal{L}',y$ then f is Θ-computable in y, because $\mathcal{L}'$ is Θ-computable. Hence $\mathrm{Ord}(f)$ is defined, and $\mathrm{Ord}(f) < \kappa_\Theta^y$. Hence $\mu_y \le \kappa_\Theta^y$.

$\varkappa^y_{\mathcal{L}} \leq \varkappa^y_{\mathcal{L}'} \leq \varkappa^y_{\Theta}$ since $\mathcal{L}'$ extends $\mathcal{L}$ and $\mathcal{L}'$ is Θ-computable. By assumption $\varkappa^y_{\mathcal{L}} = \varkappa^y_{\Theta}$. Hence $\varkappa^y_{\mathcal{L}} = \varkappa^y_{\mathcal{L}'} = \varkappa^y_{\Theta}$. By lemma 20 $\varkappa^y_{\mathcal{L}'}$ is the supremum of the lengths of the prewellorderings with domain $\subseteq A$ are recursive in $\mathcal{L}',y$. To prove that $\mu_y = \varkappa^y_{\Theta}$ it is enough to prove that for each prewellordering which is recursive in $\mathcal{L}',y$ there is a function f which is recursive in $\mathcal{L}',y$ such that $\mathrm{Ord}(f) \geq$ the length of the prewellordering.

So let $X \subseteq A^2$ be a prewellordering which is recursive in $\mathcal{L}',y$. If $x \in \mathrm{dom}(X)$ let $|x|$ be the length of that part of the prewellordering which is below x. By the second recursion theorem we find an index $\hat{e}$ such that for all $x \in \mathrm{dom}(X)$ $\lambda t\{\hat{e}\}^{\mathcal{L}'}(t,x,y)$ is total, and $|x| \leq \mathrm{Ord}(\lambda t\{\hat{e}\}^{\mathcal{L}'}(t,x,y))$. The construction goes as follows. Suppose $x \in \mathrm{dom}(X)$, and $\lambda t\{\hat{e}\}^{\mathcal{L}'}(t,x',y)$ is total and $|x'| \leq \mathrm{Ord}(\lambda t\{\hat{e}\}^{\mathcal{L}'}(t,x',y))$ for all x' such that $|x'| < |x|$. Let $f_{x'} = \lambda t\{\hat{e}\}^{\mathcal{L}}(t,x',y)$ for $|x'| < |x|$. The set $B = \{\langle e,a,z\rangle : \{e\}_{\Theta}(a) \simeq z$, and $|\{e\}_{\Theta}(a)|_{\Theta} \leq \mathrm{Ord}(f_{x'})$ for some $|x'| < |x|\}$ is recursive in $\mathcal{L}',x,y$, uniformly in x, by the construction of G, and since X is recursive in $\mathcal{L}',y$. B contains the graphs of all functions f with $\mathrm{Ord}(f) \leq \mathrm{Ord}(f_{x'})$ for some $|x'| < |x|$. By asking questions about B one can see whether or not $\lambda t\{e\}_{\Theta}(t,u)$ is total and $\mathrm{Ord}(\lambda t\{e\}_{\Theta}(t,u)) \leq \mathrm{Ord}(f_{x'})$ for some $|x'| < |x|$, for any e,u. Let $P(e,u)$ be the statement " $\lambda t\{e\}_{\Theta}(t,u)$ is total, and $\mathrm{Ord}(\lambda t\{e\}_{\Theta}(t,u)) \leq \mathrm{Ord}(f_{x'})$ for some $|x'| < |x|$ ". Then P is recursive in $\mathcal{L}',x,y$, uniformly in x. By the construction of G the function φ defined by: $\varphi(\langle e,a\rangle) \simeq \{e\}_{\Theta}(a)$ if $|\{e\}_{\Theta}(a)|_{\Theta} \leq \mathrm{Ord}(f_{x'})$ for some $|x'| < |x|$, is partial recursive in $\mathcal{L}',x,y$, uniformly in x. Let f_x be defined by: $f_x(\langle e,u\rangle) = 0$ if not $P(e,u)$, $f_x(\langle e,u\rangle) \simeq \{e\}_{\Theta}(\langle e,u\rangle,u)+1$ if $P(e,u)$. Then f_x is different from all functions f such that $\mathrm{Ord}(f) \leq \mathrm{Ord}(f_{x'})$ for some $|x'| < |x|$. f_x is recursive in $\mathcal{L}', x, y$, uniformly in x. So $\mathrm{Ord}(f_{x'}) < \mathrm{Ord}(f)$ when $|x'| < |x|$. Hence $|x| \leq \mathrm{Ord}(f_x)$ by the induction hypothesis. Let $\psi(\hat{e},t,x,y) \simeq f_x(t)$. Choose $\hat{e}$ such that $\psi(\hat{e},t,x,y) \simeq \{\hat{e}\}^{\mathcal{L}'}(t,x,y)$ for all t,x,y.

In the same way one can construct a function f such that f is recursive in $\mathcal{L}', y$, and $|x| < \mathrm{Ord}(f)$ for all $x \in \mathrm{dom}(X)$. This proves the propsition. $\square$

<u>Proposition 2</u>: $\mathrm{En}(\Theta) \subseteq \mathrm{En}(\mathcal{L}')$.

Proof: Suppose $\{e\}_\Theta(a){\downarrow}$. By proposition 1 there is an index m such that $f = \lambda t\{m\}^{\mathcal{L}'}(t,\langle e,a\rangle)$ is total, and $|\{e\}_\Theta(a)|_\Theta \leq \mathrm{Ord}(f)$. Hence $G(\langle \lambda t\{m\}^{\mathcal{L}}(t,\langle e,a\rangle),e,a\rangle) \simeq \{e\}_\Theta(a)+1$. Since $\mathcal{L}'$ is normal there is a selection function $\varphi(e,a)$ which is partial recursive in $\mathcal{L}'$ and which picks out sucn an m. Hence $\{e\}_\Theta(a) \simeq G(\langle \lambda t\{\varphi(e,a)\}^{\mathcal{L}}(t,\langle e,a\rangle),e,a\rangle) - 1$. Hence $\mathrm{En}(\Theta) \subseteq \mathrm{En}(\mathcal{L}')$. $\square$

This proves lemma 28. $\square$

<u>Definitions</u>: Θ has <u>property 1</u> if for all normal lists $\mathcal{L}$: if $\mathcal{L}$ is Θ-computable then $\mathcal{P}(\mathcal{L}) <_1 \Theta$. Θ has <u>property 2</u> if for all normal lists $\mathcal{L}$: If $\mathcal{L}$ is Θ-computable then $\mathcal{P}(\mathcal{L}) <_2 \Theta$.

<u>Lemma 29</u>: Suppose Θ is normal. Then Θ has property 1 iff Θ has property 2.

Proof: By c) in lemma 26 $\Psi <_1 \Theta \Rightarrow \Psi <_2 \Theta$. Hence if Θ has property 1 then Θ has property 2.

Suppose Θ has not property 1. Then there is a normal list $\mathcal{L}$ which is Θ-computable such that not $\mathcal{P}(\mathcal{L}) <_1 \Theta$. By a) in lemma 27 $En(\mathcal{L}) \subseteq En(\Theta)$. Hence $x(\kappa_{\mathcal{L}}^{x} = \kappa_{\Theta}^{x})$. By lemma 28 there is a normal list $\mathcal{L}'$ such that $\mathcal{P}(\mathcal{L}') \sim \Theta$. By b) of lemma 27 $\mathcal{L}'$ is Θ-computable. $\mathcal{P}(\mathcal{L}') <_2 \Theta$ is not true. Hence Θ has not property 2. $\square$

__Definition__: Θ is __Mahlo__ if Θ has property 1.

__Theorem 10__: Let Θ be normal. Then Θ is not Mahlo iff there is a normal list $\mathcal{L}$ such that $\mathcal{P}(\mathcal{L}) \sim \Theta$.

Proof: Suppose Θ is not Mahlo. By lemma 29 Θ has not property 2. Hence there is a normal list $\mathcal{L}$ such that $\mathcal{L}$ is Θ-computable and not $\mathcal{P}(\mathcal{L}) <_2 \Theta$. Hence $En(\mathcal{L}) = En(\Theta)$, and $\mathcal{P}(\mathcal{L}) \sim \Theta$.

Suppose that there is a normal list $\mathcal{L}$ such that $\mathcal{P}(\mathcal{L}) \sim \Theta$. $\mathcal{L}$ is Θ-computable by b) of lemma 27. Hence Θ has not property 2. By lemma 29 Θ is not Mahlo. $\square$

§ 9 MORE ABOUT MAHLONESS

In ordinal recursion the notion of Mahloness is defined in the following way: An ordinal τ is Mahlo if τ is recursively regular, and all normal functions π which are τ-recursive in constants less that τ have a recursively regular fixpoint less that τ. (Definition 4.2 (b) in [1]). In § 8 the notion of Mahloness was defined in another way. The purpose of this chapter is to prove that the definition in § 8 is a natural generalization of the definition above.

To see this let us regard normal computation theories $(\Theta, ||_\Theta)$ with domain ω, i.e. ω is strongly Θ-finite and $||_\Theta$ is a Θ-norm. Since the domain is ω $\kappa_\Theta^x = \kappa_\Theta$ for all x in the domain. We define property 1 as in § 8 : Θ has property 1 if for all normal lists $\mathcal{L}$: If $\mathcal{L}$ is Θ-computable then $\kappa^{\mathcal{L}} < \kappa_\Theta$.

Below we introduce ordinal recursion for $(\Theta, ||_\Theta)$. Let $\tau \leq \kappa_\Theta$. Let $\Theta_\tau = \{(e,a,r) : \{e\}_\Theta(a) \simeq r$ and $|\{e\}_\Theta(a)|_\Theta < \tau\}$. Let $||_{\Theta\tau}$ be the restriction of $||_\Theta$ to Θ_τ. Then $||_{\Theta\tau}$ is a mapping of Θ_τ onto τ. τ is Θ-regular if $(\Theta_\tau, ||_{\Theta\tau})$ is a normal computation theory on ω.

Let $\tau \leq \kappa_\Theta$. A relation R is Θ_τ-semicomputable if there is an index e such that for all a : $R(a) \iff |\{e\}_\Theta(a)| < \tau$. Let π be a partial function from τ^n to τ. π is Θ_τ-computable if the set $\{(x_1 \ldots x_n, y) : |x_1|_\Theta, \ldots, |x_n|_\Theta, |y|_\Theta < \tau$ and $\pi(|x_1|_\Theta, \ldots, |x_n|_\Theta) \simeq |y|_\Theta\}$ is Θ_τ-semicomputable. π is Θ-computable if π is Θ_κ-computable, where $\kappa = \kappa_\Theta$

A function π from τ to τ is normal if it is strictly increasing and continuous at limits. An ordinal $\nu < \tau$ is a fixed point for π if $\pi(\nu) = \nu$. $\tau \leq \kappa_\Theta$ is Θ-Mahlo if τ is Θ-regular and all normal Θ_τ- computable functions have a Θ-regular fixed point less than τ. Θ is Mahlo if κ_Θ is Θ-Mahlo.

When $(\Theta, ||_\Theta)$ is the computation theory obtained from a normal list $\mathcal{L}$, we will write $\mathcal{L}$-regular, $\mathcal{L}_\tau$-computable, $\mathcal{L}$-computable instead of Θ-regular, Θ_τ- computable, Θ-computable.

In ordinary recursion theory an ordinal τ is said to be recursively regular if the following is true: If $\nu < \tau$, and π is a partial function which is τ-recursive in constants less that τ, and $\pi(\xi)\downarrow$ for all $\xi < \nu$, then there is an ordinal $\tau' < \tau$ such that $\pi(\xi) < \tau'$ for all $\xi < \nu$. (Definition 3.4 in [1])

In view of this definition we prove the following lemma.

Lemma 30: If $(\Theta, ||_\Theta)$ is a normal computation theory on ω then the following is true: If $\nu < \kappa_\Theta$, and π is a partial function which is Θ-computable, and $\pi(\xi)\downarrow$ for all $\xi < \nu$, then there is an ordinal $\tau' < \kappa_\Theta$ such that $\pi(\xi) < \tau'$ for all $\xi < \nu$.

Proof: Let $\nu < \kappa_\Theta$, and let π be a partial function which is Θ-computable, such that $\pi(\xi)\downarrow$ for all $\xi < \nu$. The set $\{(x,y) : |x|_\Theta, |y|_\Theta < \kappa_\Theta \text{ and } \pi(|x|_\Theta) \simeq |y|_\Theta\}$ is Θ-semicomputable. There is a Θ-computable selection function $\varphi(x)$ such that if $\exists y(\pi(|x|_\Theta) \simeq |y|_\Theta)$ then $\varphi(x)\downarrow$, and $\pi(|x|_\Theta) \simeq |\varphi(x)|_\Theta$. Choose a w such that $|w|_\Theta = \nu$. Choose an index e as follows: To compute $\{e\}_\Theta(w)$ take each x such that $|x|_\Theta < |w|_\Theta$, find $\varphi(x)$, and compute this computation, i.e. if $\varphi(x) = \langle e', a' \rangle$, compute $\{e'\}_\Theta(a')$. Then $\{e\}_\Theta(w)\downarrow$, and $|\varphi(x)|_\Theta < |\{e\}_\Theta(w)|$ for all x such that $|x|_\Theta < |w|_\Theta$. Let $\tau' = |\{e\}_\Theta(w)|$. Then $\tau' < \kappa_\Theta$, and $\pi(\xi) < \tau'$ for all $\xi < \nu$. □

<u>Remark</u>: From lemma 30 one can prove that κ_Θ is recursively regular. The following result is also true: If τ is recursively regular, and τ is projectible to ω (i.e. there is a one-one mapping π from τ to ω which is τ-recursive in constants less than τ) then there is a normal computation theory $(\Theta, ||_\Theta)$ on ω such that $\kappa_\Theta = \tau$.

<u>Lemma 31</u>: If $\mathcal{L}$ is a normal list then $\kappa^{\mathcal{L}}$ is the least $\mathcal{L}$-regular ordinal.

Proof: Obviously all Θ-regular ordinals are limit ordinals $> \omega$ for any normal Θ. Let τ be a limit ordinal such that $\omega < \tau < \kappa^{\mathcal{L}}$. Since $\tau < \kappa^{\mathcal{L}}$ the induction which generates the convergent $\mathcal{L}$-computations does not stop at the stage τ. Hence there are $\mathcal{L}$-computations with length τ. Since τ is a limit ordinal such computations are applications of a functional to a function $\lambda x\{e\}^{\mathcal{L}}(x,a)$, where $\lambda x\{e\}^{\mathcal{L}}(x,a)$ is total, and $\tau = \sup\{|\{e\}^{\mathcal{L}}(x,a)| +1 : x \in \omega\}$. Let π be defined by: $\pi(n) = |\{e\}^{\mathcal{L}}(n,a)|^{\mathcal{L}}$. Then $\tau = \sup\{\pi(n) : n \in \omega\}$. If τ was $\mathcal{L}$-regular then π would be $\mathcal{L}_\tau$-computable. By lemma 30 $\sup\{\pi(n) : n \in \omega\} < \tau$, a contradiction. Hence τ is not $\mathcal{L}$-regular. $\square$

<u>Lemma 32</u>: If $(\Theta, ||_\Theta)$ is a normal computation theory on ω, and $\mathcal{L}$ is a normal Θ-computable list, then there is a Θ-computable normal function π which has no Θ-regular fixed points less than $\kappa^{\mathcal{L}}$.

Proof: Let $(\Theta, ||_\Theta)$ and $\mathcal{L}$ be as in the hypothesis. Since $\mathcal{L}$ is Θ-computable there is an index t such that for all x: $x \in C^{\mathcal{L}} \iff \langle t,x\rangle \in C_\Theta$. ($C^{\mathcal{L}}$ is the set of convergent $\mathcal{L}$-computations, C_Θ is the set of convergent Θ-computations.) If $\nu < \kappa_\Theta$ then the set $\{x : |x|^{\mathcal{L}} < \nu\}$ is Θ-computable. There is an ordinal $\mu < \kappa_\Theta$ such that for all $x : |x|^{\mathcal{L}} < \nu \implies |\langle t,x\rangle|_\Theta < \mu$. For the function π defined by: $\pi(|x|^{\mathcal{L}}) \simeq |\langle t,x\rangle|_\Theta$ is Θ-computable. By lemma 30 $\sup\{\pi(\eta) : \eta < \nu\} < \kappa_\Theta$. In the same way one can prove that if $\nu < \kappa_\Theta$ then there is an ordinal $\mu < \kappa_\Theta$ such that for all $x : |\langle t,x\rangle|_\Theta < \nu \implies |x|^{\mathcal{L}} < \mu$.

Let π be defined as follows: $\pi(0) = 0$, π is continuous at limits. If ν is a successor ordinal, let $\pi(\nu)$ be the least ordinal μ such that:

i) $\pi(\nu-1) < \mu$

ii) for all $x : |x|^{\mathcal{L}} < \nu \implies |\langle t,x\rangle|_\Theta < \mu$,

iii) for all $x : |\langle t,x\rangle|_\Theta < \nu \implies |x|^{\mathcal{L}} < \mu$.

By the discussion above $\pi(\nu)$ is defined, and $\pi(\nu) < \kappa_\Theta$ when $\nu < \kappa_\Theta$. By the construction π is normal. By an application of the second recursion theorem one can prove that π is Θ-computable.

It remains to prove that π has no Θ-regular fixed points less than $\kappa^{\mathcal{L}}$. Suppose $\tau < \kappa^{\mathcal{L}}$, $\lim \tau$ and $\pi(\tau) = \tau$. Since $\tau < \kappa^{\mathcal{L}}$ and $\lim \tau$ there is an index e and a list a such that $\lambda x\{e\}^{\mathcal{L}}(x,a)$ is total, $|\{e\}^{\mathcal{L}}(x,a)|^{\mathcal{L}} < \tau$ for all x, and $\sup\{|\{e\}^{\mathcal{L}}(x,a)|^{\mathcal{L}} : x \in \omega\} = \tau$. By ii) $|\langle t,\langle e,x,a\rangle\rangle|_\Theta < \pi(\tau)$ for all x, hence $|\langle t,\langle e,x,a\rangle\rangle|_\Theta < \tau$ for all x. $\sup\{|\langle t,\langle e,x,a\rangle\rangle|_\Theta : x \in \omega\} = \tau$, for suppose that this supremum was equal to $\tau' < \tau$. Then by iii) $|\{e\}^{\mathcal{L}}(x,a)|^{\mathcal{L}} < \pi(\tau'+1)$ for all x. Now $\pi(\tau'+1) < \pi(\tau) = \tau$. So this is contrary to the fact that $\sup\{|\{e\}^{\mathcal{L}}(x,a)|^{\mathcal{L}} : x \in \omega\} = \tau$. If τ was Θ-regular then the function ρ defined by: $\rho(x) = |\langle t,\langle e,x,a\rangle\rangle|_\Theta$ for $x \in \omega$, would be Θ_τ-computable. By lemma 30 $\sup\{\rho(x) : x \in \omega\} < \tau$, contrary to what is proved above. So τ is not Θ-regular. $\square$

In the proof of the next lemma we need the ordinal function ρ defined below.

Let $(\Theta, ||_\Theta)$ be a normal computation theory on ω. Since it is a computation theory the operations mentioned in the beginning of § 6, such substitution, primitive recursion etc. are allowed. Let us regard substitution: There is a Θ-computable mapping $g_1(e,f,n)$ such that for all e, f, a, x: $\{g_1(e,f,n)\}_\Theta(a) \simeq x \iff \exists u\,[\{e\}_\Theta(a) \simeq u$ and $\{f\}_\Theta(u,a) \simeq x]$, and if $\{g_1(e,f,n)\}_\Theta(a) \simeq x$ then for some u such that $\{e\}_\Theta(a) \simeq u$ and $\{f\}_\Theta(u,a) \simeq x$: $|\{e\}_\Theta(a)|$, $|\{f\}_\Theta(u,a)| < |\{g_1(e,f,n)\}_\Theta(a)|$. This operation has a finitary character, i.e. there are only finitely many natural predecessors of the computation $\{g_1(e,f,n)\}_\Theta(a)$, namely $\{e\}_\Theta(a)$ and $\{f\}_\Theta(u,a)$. The other operations mentioned in § 6 are also finitary.

Since $(\Theta, ||_\Theta)$ is normal there is a partial Θ-computable function p such that $p(x,y) \simeq 0$ if $x \in C_\Theta$ and $|x|_\Theta \leq |y|_\Theta$. This can also be regarded as a finitary operation. There is one natural predecessor of $\{\hat{p}\}_\Theta(x,y)$, namely x if $x \in C_\Theta$ and $|x|_\Theta \leq |y|_\Theta$, y if $|x|_\Theta > |y|_\Theta$. ($\hat{p}$ is a Θ-index for p)

Since ω is strongly Θ-finite the functional E_ω is weakly Θ-finite, where E_ω is defined by: $E_\omega(\varphi) \simeq 0$ if $\exists x\, \varphi(x) \simeq 0$, $E_\omega(\varphi) \simeq 1$ if $\forall x\, \exists y \neq 0\; \varphi(x) \simeq y$. There is a Θ-computable mapping $g_2(n)$ such that $\{g_2(n)\}_\Theta(e,a) \simeq E_\omega(\lambda x\{e\}_\Theta(x,a))$ for all e, a (the list a has length n). If $\{g(n)\}_\Theta(e,a) \simeq 0$ then the natural predecessors of $\{g(n)\}_\Theta(e,a)$ are those computations $\{e\}_\Theta(x,a)$ such that $\{e\}_\Theta(x,a) \simeq 0$ and $\{e\}_\Theta(x,a)$ has minimal length. If $\{g(n)\}_\Theta(e,a) \simeq 1$ then the natural predecessors of $\{g(n)\}_\Theta(e,a)$ are $\{e\}_\Theta(x,a)$ for all x. So this operation is not finitary.

Let $\nu < \varkappa_\Theta$. There is an ordinal $\mu < \varkappa_\Theta$ such that the following is true:

Substitution: If for some u $\{e\}_\Theta(a) \simeq u$ and $\{f\}_\Theta(u,a) \simeq x$, and $|\{e\}_\Theta(a)|$, $|\{f\}_\Theta(u,a)| < \nu$ then $|\{g_1(e,f,u)\}_\Theta(a)| < \mu$.

p : If $|x|_\Theta < \nu$ or $|y|_\Theta < \nu$ then $|\{\hat{p}\}_\Theta(x,y)| < \mu$.

E_ω : If for some x $\{e\}_\Theta(x,a) \simeq 0$ and $|\{e\}_\Theta(x,a) < \nu$ then $|\{g_2(n)\}_\Theta(e,a)| < \mu$. If for all x there is a $y \neq 0$ such that $\{e\}_\Theta(x,a) \simeq y$ and $|\{e\}_\Theta(x,a)| < \nu$ then $|\{g_2(n)\}_\Theta(e,a)| < \mu$.

In addition there are clauses for the other operations mentioned in § 6.

There is a $\mu < \varkappa_\Theta$ which satisfies these conditions. Otherwise we would obtain a contradiction to lemma 30. Let $\rho(\nu)$ be the least ordinal μ which satisfies the conditions above. Then $\rho(\nu) < \varkappa_\Theta$ when $\nu < \varkappa_\Theta$, and ρ is Θ-computable.

ρ is not necessarily continuous at limit ordinals. If $(\Theta, ||_\Theta)$ is the computation theory obtained from a normal list $\mathcal{L}$ it can be proved that $\rho(\nu) <$ the least limit ordinal $> \nu$. If $E_\omega(\lambda x\{e\}^{\mathcal{L}}(x,a))$ $\simeq 1$ and $\sup\{|\{e\}^{\mathcal{L}}(x,a)|^{\mathcal{L}} + 1 : x \in \omega\} = \lambda$, and $\lim \lambda$, then $\sup\{\rho(\nu) : \nu < \lambda\} = \lambda < \rho(\lambda)$. Hence ρ is not continuous at λ.

Suppose $\tau < \varkappa_\Theta$ is a limit ordinal such that $\nu < \tau \Rightarrow \rho(\nu) < \tau$. If $\rho(\tau) = \tau$ then $(\Theta_\tau, ||_{\Theta_\tau})$ is a normal computation theory on ω. $\rho(\tau) = \tau$ iff the following is true: For all e, a if $|\{e\}_\Theta(x,a)|$ $< \tau$ for all x, then $|\{g_2(n)\}_\Theta(e,a)| < \tau$.

<u>Lemma 33</u>: If $(\Theta, ||_\Theta)$ is a normal computation theory on ω, and π is a normal Θ-computable function, then there is a normal Θ-computable list $\mathcal{L}$ such that $\varkappa^{\mathcal{L}}$ is Θ-regular and a fixed point for π.

Proof: There is an index e_0 such that $\{e_0\}_\Theta(e)\downarrow$ iff $\lambda x\{e\}_\Theta(x)$ is total, in which case $|\{e\}_\Theta(x,a)|_\Theta < |\{e_0\}_\Theta(e)|_\Theta$ for all x. If f is a Θ-computable total function let $\mathrm{Ord}(f) = \inf\{|\{e_0\}_\Theta(e)|_\Theta : f = \lambda x\{e\}_\Theta(x)\}$.

Let the functional F be defined as follows. If f is Θ-computable and total, let $\nu = \mathrm{Ord}(f)$, and let $\mu = \sup(\pi(\nu),\rho(\nu))$. Let

$$F(\langle f,n,0\rangle) = \begin{cases} 0 & \text{if } n = \langle e,a,y\rangle,\ \{e\}_\Theta(a) \simeq y,\ |\{e\}_\Theta(a)|_\Theta < \nu \\ 1 & \text{otherwise} \end{cases}$$

$$F(\langle f,n,1\rangle) = \begin{cases} 0 & \text{if } n = \langle e,a,y\rangle,\ \{e\}_\Theta(a) \simeq y,\ |\{e\}_\Theta(a)|_\Theta < \mu \\ 1 & \text{otherwise} \end{cases}$$

If g is not Θ-computable, or g is not of the form $\langle f,n,0\rangle$ or $\langle f,n,1\rangle$, let $F(g) = 1$. Then F is weakly Θ-computable. Hence the list $\mathcal{L} = E,F$ is Θ-computable.

If f is recursive in $\mathcal{L}$ and total then f is Θ-computable, hence $\mathrm{Ord}(f)$ is defined. Let $\lambda = \sup\{\mathrm{Ord}(f) : f$ is total and recursive in $\mathcal{L}\}$.

Suppose $f = \lambda x\{e\}^{\mathcal{L}}(x)$ is total. Let $\nu = \mathrm{Ord}(f)$. $\lambda n\, F(\langle f,n,0\rangle)$ is the characteristic function of the set $B_e = \{\langle e',a,y\rangle : \{e'\}_\Theta(a) \simeq y$ and $|\{e'\}_\Theta(a)|_\Theta < \nu\}$. The set $C_e = \{\langle e',a,y\rangle : \{e'\}_\Theta(a) \simeq y$ and $|\{e'\}_\Theta(a)|_\Theta < \mu\}$ has the characteristic function $\lambda n\, F(\langle f,n,1\rangle)$ ($\mu = \sup(\pi(\nu),\rho(\nu))$. Hence B_e and C_e are recursive in $\mathcal{L}$, and $\mathcal{L}$-indexes for the characteristic functions of B_e and C_e can be found uniformly from e.

Let $\hat{p}$ be the index for the function p mentioned in the construction of ρ. If $\inf(|x|_\Theta, |y|_\Theta) < \nu$ then $|\{\hat{p}\}_\Theta(x,y)|_\Theta < \rho(\nu)$ by the construction of ρ. Hence $|\{\hat{p}\}_\Theta(x,y)|_\Theta < \mu$. The set $\{(x,y) : |x|_\Theta < \nu$ or $|y|_\Theta < \nu$, and $\langle \hat{p},x,y,0\rangle \in C_e\}$ is a prewellordering of length ν. It is recursive in $\mathscr{L}$ since B_e and C_e are recursive in $\mathscr{L}$. By lemma 20 each prewellordering which is recursive in $\mathscr{L}$ has length $< \varkappa^{\mathscr{L}}$. Hence $\nu < \varkappa^{\mathscr{L}}$, and $\lambda \leq \varkappa^{\mathscr{L}}$.

By uniformity there is a primitive recursive mapping g such that $\{g(e)\}^{\mathscr{L}}(x,y) \simeq 0$ if $|x|_{\Theta} < \nu$ and $|x|_{\Theta} \leq |y|_{\Theta}$, $\{g(e)\}^{\mathscr{L}}(x,y) \simeq 1$ if $|x|_{\Theta} > |y|_{\Theta}$ and $|y|_{\Theta} < \nu$.

There is an index e_1 such that $\{e_1\}^{\mathscr{L}}(e)\downarrow$ iff $\lambda x\{e\}^{\mathscr{L}}(x)$ is total, in which case $|\{e\}^{\mathscr{L}}(x)|^{\mathscr{L}} < |\{e_1\}^{\mathscr{L}}(e)|^{\mathscr{L}}$ for all x. If f is total and recursive in $\mathscr{L}$ let $\mathrm{Ord}^{\mathscr{L}}(f) = \inf\{|\{e_1\}^{\mathscr{L}}(e)|^{\mathscr{L}} : f = \lambda x\{e\}^{\mathscr{L}}(x)\}$.

If $\mu < \varkappa^{\mathscr{L}}$ then by a simple diagonalization method one can construct a function f which is total and recursive in $\mathscr{L}$, and f is different from all functions f' with $\mathrm{Ord}^{\mathscr{L}}(f') < \mu$. Hence $\mathrm{Ord}^{\mathscr{L}}(f) \geq \mu$, and $\sup\{\mathrm{Ord}^{\mathscr{L}}(f) : f$ is recursive in $\mathscr{L}\} = \varkappa^{\mathscr{L}}$.

If $\mu < \varkappa^{\mathscr{L}}$ let $\mathscr{F}_{\mu} = \{f : \mathrm{Ord}^{\mathscr{L}}(f) < \mu\}$. Let $\nu = \sup\{\mathrm{Ord}(f) : f \in \mathscr{F}_{\mu}\}$. Claim: $\nu < \lambda$.

Proof: Let $D = \{e : |\{e_1\}^{\mathscr{L}}(e)| < \mu\}$. Then D is recursive in $\mathscr{L}$, and $f \in \mathscr{F}_{\mu}$ iff $f = \lambda x\{e\}^{\mathscr{L}}(x)$ for some $e \in D$. The set $\bigcup_{e\in D} B_e$ is recursive in $\mathscr{L}$ since B_e is recursive in $\mathscr{L}$ uniformly in e. $\bigcup_{e\in D} B_e = \{\langle e',a,y\rangle : \{e'\}_{\Theta}(a) \simeq y$ and $|\{e'\}_{\Theta}(a)|_{\Theta} < \nu\}$.
If f is a total Θ-computable function such that $\mathrm{Ord}(f) \leq \nu$ then there is an index e such that $f = \lambda x\{e\}_{\Theta}(x)$, and $\forall x \exists y \langle e,x,y\rangle \in \bigcup_{e\in D} B_e$. Hence the set $E = \{e : \forall x \exists y \langle e,x,y\rangle \in \bigcup_{e\in D} B_e\}$ contains Θ-indexes for all total Θ-computable functions f with $\mathrm{Ord}(f) \leq \nu$, in particular E contains Θ-indexes for all $f \in \mathscr{F}_{\mu}$. E is recursive in $\mathscr{L}$. By a diagonalization method one can construct a function f' which is recursive in $\mathscr{L}$, and such that f' is different from all total Θ-computable functions with Θ-index

in E. Hence $\mathrm{Ord}(f') > \nu$, and $\nu < \lambda$.

Let the ordinal function σ be defined by: If $\nu < \lambda$ let $\sigma(\nu) = \inf\{\mathrm{Ord}^{\mathcal{L}}(f) : \mathrm{Ord}(f) > \nu\}$. Then $\sup\{\sigma(\nu) : \nu < \lambda\} = \varkappa^{\mathcal{L}}$, for suppose $\sup\{\sigma(\nu) : \nu < \lambda\} = \mu < \varkappa^{\mathcal{L}}$. Then for all $\nu < \lambda$ there is an f such that $\mathrm{Ord}(f) > \nu$ and $\mathrm{Ord}^{\mathcal{L}}(f) < \mu$. Hence $\sup\{\mathrm{Ord}(f) : \mathrm{Ord}^{\mathcal{L}}(f) < \mu\} = \lambda$, contrary to the claim.

To prove that σ is recursive in $\mathcal{L}$ it is enough to prove that R is recursively enumerable in $\mathcal{L}$, where $R = \{(x,y) : x,y \in C^{\mathcal{L}}$ and $\sigma(|x|^{\mathcal{L}}) = |y|^{\mathcal{L}}\}$. $(x,y) \in R \iff x,y \in C^{\mathcal{L}}$ and $\exists e\,[\lambda x\{e\}^{\mathcal{L}}(x)$ is total and $|x|^{\mathcal{L}} < \mathrm{Ord}(f)$ and $|y|^{\mathcal{L}} = \mathrm{Ord}^{\mathcal{L}}(f)$ and $\forall e'(\mathrm{Ord}^{\mathcal{L}}(f') < \mathrm{Ord}^{\mathcal{L}}(f) \Longrightarrow \mathrm{Ord}(f') \leq |x|^{\mathcal{L}})]$ $(f = \lambda x\{e\}^{\mathcal{L}}(x)$, $f' = \lambda x\{e'\}^{\mathcal{L}}(x)$.) "$\lambda x\{e\}^{\mathcal{L}}(x)$ is total" can be expressed by "$\{e_1\}^{\mathcal{L}}(e)\downarrow$". "$|y|^{\mathcal{L}} = \mathrm{Ord}^{\mathcal{L}}(f)$" can be expressed by "$|y|^{\mathcal{L}} = |\{e_1\}^{\mathcal{L}}(e)|^{\mathcal{L}}$", which is recursively enumerable in $\mathcal{L}$. "$\mathrm{Ord}^{\mathcal{L}}(f') < \mathrm{Ord}^{\mathcal{L}}(f)$" can be expressed by "$|\{e_1\}^{\mathcal{L}}(e')|^{\mathcal{L}} < |\{e_1\}^{\mathcal{L}}(e)|^{\mathcal{L}}$". It remains to find expressions for "$|x|^{\mathcal{L}} < \mathrm{Ord}(f)$" and "$\mathrm{Ord}(f') \leq |x|^{\mathcal{L}}$". This is done in the next section.

Let $f = \lambda x\{e\}^{\mathcal{L}}(x)$ be total, and $\mathrm{Ord}(f) = \nu$. Then there is an index $\hat{e}$ such that $f = \lambda x\{\hat{e}\}_{\Theta}(x)$, $|\{\hat{e}\}_{\Theta}(x)|_{\Theta} < \nu$ for all x, and $\nu = |\{e_0\}_{\Theta}(\hat{e})|_{\Theta}$. Also $\forall x \exists y\,(\langle \hat{e},x,y\rangle \in B_e)$. We can find such an index $\hat{e}$ by asking questions about B_e and C_e, which are recursive in $\mathcal{L}$, uniformly in e. There is a function t which is partial recursive in $\mathcal{L}$ such that if $\lambda x\{e\}^{\mathcal{L}}(x)$ is total then $t(e)$ is defined, and $t(e)$ is such a Θ-index for f. An $\mathcal{L}$-index for t can be constructed from the primitive recursive mapping g. There is also a primitive recursive mapping h such that if $f = \lambda x\{e\}^{\mathcal{L}}(x)$ is total and $\mathrm{Ord}(f) = \nu$ then

$\{h(e)\}^{\mathcal{L}}(x,y) \simeq 0$ if $|x|_{\Theta} \leq \nu$ and $|x|_{\Theta} \leq |y|^{\mathcal{L}}$,

$\{h(e)\}^{\mathcal{L}}(x,y) \simeq 1$ if $|x|_{\Theta} \leq \nu$ and $|x|_{\Theta} > |y|^{\mathcal{L}}$.

" $|x|^{\mathcal{L}} < \mathrm{Ord}(f)$ " can be expressed by " $\{h(e)\}^{\mathcal{L}}(\langle e_0, e\rangle, x) \simeq 1$ ", " $\mathrm{Ord}(f') \leq |x|^{\mathcal{L}}$ " can be expressed by " $\{h(e)\}^{\mathcal{L}}(\langle e_0, e'\rangle, x) \simeq 0$ ". This proves that R is recursively enumerable in $\mathcal{L}$, and hence that σ is $\mathcal{L}$-recursive.

As mentioned before $\sup\{\sigma(\nu) : \nu < \lambda\} = \kappa^{\mathcal{L}}$, and $\lambda \leq \kappa^{\mathcal{L}}$. It follows from lemma 30 that $\lambda = \kappa^{\mathcal{L}}$.

Nest we prove that if $\nu < \kappa^{\mathcal{L}}$ then $\pi(\nu) < \kappa^{\mathcal{L}}$ and $\rho(\nu) < \kappa^{\mathcal{L}}$. Let $\nu < \kappa^{\mathcal{L}}$. Choose e such that $f = \lambda x\{e\}^{\mathcal{L}}(x)$ is total, and $\mathrm{Ord}^{\mathcal{L}}(f) = \nu' \geq \nu$. Let $\mu = \sup(\pi(\nu'), \rho(\nu'))$. C_e is recursive in $\mathcal{L}$. From C_e one can construct a function f' which is total and recursive in $\mathcal{L}$, and f' is different from all functions f'' with $\mathrm{Ord}(f'') \leq \mu$. Hence $\mu < \mathrm{Ord}(f')$. Since $\mathrm{Ord}(f') < \lambda = \kappa^{\mathcal{L}}$, $\mu < \kappa^{\mathcal{L}}$. Hence $\rho(\nu') < \kappa^{\mathcal{L}}$ and $\pi(\nu') < \kappa^{\mathcal{L}}$, and $\rho(\nu) < \kappa^{\mathcal{L}}$ and $\pi(\nu) < \kappa^{\mathcal{L}}$ since $\nu \leq \nu'$. This proves that $\kappa^{\mathcal{L}}$ is a fixed point for π.

To prove that $\kappa^{\mathcal{L}}$ is Θ-regular it is enough to prove that if $|\{e\}_\Theta(x,a)|_\Theta < \kappa^{\mathcal{L}}$ for all x, then there is an ordinal $\mu < \kappa$ such that $|\{e\}_\Theta(x,a)|_\Theta < \mu$ for all x. So suppose $|\{e\}_\Theta(x,a)|_\Theta < \kappa^{\mathcal{L}}$ for all x. From the functions defined earlier in this proof one can construct an index e' such that $\{e\}_\Theta(x,a) \simeq \{e'\}^{\mathcal{L}}(x,a)$ for all x, and $|\{e\}_\Theta(x,a)|_\Theta < |\{e'\}^{\mathcal{L}}(x,a)|^{\mathcal{L}}$ for all x. There is a $\mu < \kappa^{\mathcal{L}}$ such that $|\{e'\}^{\mathcal{L}}(x,a)|^{\mathcal{L}} < \mu$ for all x. (Let μ be the length of the computation $E(\lambda x\{e'\}^{\mathcal{L}}(x,a))$.) Hence $|\{e\}_\Theta(x,a)|_\Theta < \mu$ for all x. □

<u>Theorem 11</u>: Let $(\Theta, ||_\Theta)$ be a normal computation theory on ω. Then Θ is Mahlo iff Θ has property 1.

Proof: Suppose Θ is Mahlo. Let $\mathcal{L}$ be a normal Θ- computable list. By lemma 32 there is a normal Θ-computable ordinal function π which has no Θ-regular fixed points less than $\kappa^{\mathcal{L}}$. Since Θ is Mahlo there is a Θ-regular fixed point for π less that κ_{Θ}. Hence $\kappa^{\mathcal{L}} < \kappa_{\Theta}$, and Θ has property 1.

Suppose Θ has property 1. Let π be a normal Θ-computable ordinal function. By lemma 33 there is a normal Θ-computable list $\mathcal{L}$ such that $\kappa^{\mathcal{L}}$ is Θ-regular and a fixed point for π. Since Θ has property 1 $\kappa^{\mathcal{L}} < \kappa_{\Theta}$. Hence π has a Θ-regular fixed point less that κ_{Θ}. Hence Θ is Hahlo.

$\square$

§ 10 A FINAL COMMENT CONCERNING THE TWO TYPES

Some of the results in this paper depend on the fact that the universe of the computation domain consists of two types, some results are independent of this. Below follows a short review of the paper, where special attention is paid to this dependence.

Let $\mathfrak{S}$ be the structure $(A, N, +1, M, K, L)$ where $N \subseteq A$ is a copy of the natural numbers with successor function $+1$, M is a pairing function on A with inverse functions K and L. This structure is more general than the structure in § 1 since the subindividuals are not given a priori. Given a list $\mathcal{L}$ of relations, functions and functionals one can develop recursion theory on $\mathfrak{S}$ relative to $\mathcal{L}$, as it is done in § 2. Theorem 1 is still true. The list $\mathcal{L}$ is normal if the equality relation on A is recursive in $\mathcal{L}$, and A is weakly finite. Theorems 2 and 3 can be proved as in § 4. In the proof of theorem 4 we used the fact that countably many elements in A can be coded as one element in A in an $\mathcal{L}$-recursive way. This is true because the computation domain consists of two types. The same fact is used in the proofs of theorems 5 and 6. The type structure is also essential in the proof of theorem 7. The following facts are used: If $X \subseteq A$ is a set indexed by S then all the elements in X can be coded by one element in A in an $\mathcal{L}$-recursive way. The set of prewellorderings with domain $\subseteq S$ is recursive in $\mathcal{L}$. (The last statement is a corollary of the first.)

One can define the notion of a normal computation theory on $\mathfrak{S}$ almost as in § 6 and § 8. Lemma 20 is still true. To prove lemma 21 we essentially use the fact that the relation " x is a prewellordering with domain $\subseteq S$ " is Θ-computable. Hence the type structure is needed. Lemma 22 is independent of the type structure. In the proof of lemma 23 we use the fact that S is strongly finite, which is an assumption for a normal computation theory. Hence this lemma is independent of the type structure. Lemmas 24 and 25 are also independent. In the proof of theorem 8 we use the fact that the relation " x is a prewellordering with domain $\subseteq S$ " is Θ-computable. So the type structure is needed. Lemma 21, theorem 8 and the corollary of theorem 8 are used in the proof of theorem 9. Theorem 10 is independent of the type structure.

References:

[1] D. Cenzer. Ordinal recursion and inductive definitions. Proceedings of the 1972 Oslo Symposium.

[2] J.E. Fenstad. On axiomatizing recursion theory. Proceedings of the 1972 Oslo Symposium.

[3] J.E. Fenstad. Computation theories: An axiomatic approach to recursion on general structures. Proceedings of the Kiel Summer Institute in Logic 1974.

[4] R.O. Gandy. General recursive functionals of finite type and hierarchies of functionals. Ann. Fac. Sci. Univ. Clermont-Ferrand, No.35 (1967).

[5] T. Grilliot. Hierarchies based on objects of finite type. Journal of Symbolic Logic 34 (1969).

[6] T. Grilliot. Selection functions for recursive functionals. Notre Dame Journal of Formal Logic, 10 (1969).

[7] L.A. Harrington. Contributions to recursion theory on higher types. Ph.D. Thesis. Massachusetts Institute of Technology (1973).

[8] L.A. Harrington, A.S. Kechris. Classifying and characterizing abstract classes of relations. Massachusetts Institute of Technology.

[9] L.A. Harrington, A.S. Kechris, S.G. Simpson. 1-envelopes of type 2 objects.

[10] L.A. Harrington, D.B. McQueen. Selection in abstract recursion theory.

[11] A.S. Kechris. The structure of envelopes: A survey of recursion theory in higher types. Massachusetts Institute of Technology (1973).

[12] S.C. Kleene. Recursive functionals and quantifiers of finite types I (Trans. Amer. Math. Soc. 91 (1959)), and II (Trans. Amer. Math. Soc. 108 (1963)).

[13] D.B. McQueen. Post's problem for recursion in higher types. Ph.D. Thesis. Massachusetts Institute of Technology (1972).

[14] Y.N. Moschovakis. Hyperanalytic predicates. Trans. Amer. Math. Soc. 129 (1967).

[15] Y.N. Moschovakis. Abstract first order computability I, II. Trans. Amer. Math. Soc. 138 (1969).

[16] Y.N. Moschovakis. Axioms for computation theories-first draft. Logic Colloquium 69.

[17] Y.N. Moschovakis. Structural characterizations of classes of relations. Proceedings of the 1972 Oslo Symposium.

[18] Y.N. Moschovakis. Elementary induction on abstract structures. North-Holland Publishing Company 1974.

[19] D. Normann. Imbedding of higher type theories. Preprint Oslo 1974.

[20] G.E. Sacks. The 1-section of a type n object. Proceedings of the 1972 Oslo Symposium.

[21] G.E. Sacks. The k-section of a type n object .

[22] J.R. Shoenfield. A hierarchy based on a type two object. Trans. Amer. Math. Soc. 134 (1968).